

**ANALOGICAL PROBLEM EVOLUTION IN BIOLOGICALLY INSPIRED
DESIGN**

A Thesis
Presented to
The Academic Faculty

By

Michael Helms

In Partial Fulfillment
Of the Requirements for the Degree
Doctor of Philosophy in Computer Science

Georgia Institute of Technology

December, 2013

COPYRIGHT 2013 BY MICHAEL E. HELMS

**ANALOGICAL PROBLEM EVOLUTION IN BIOLOGICALLY INSPIRED
DESIGN**

Approved By:

Dr. Ashok K. Goel, Advisor
School of Interactive Computing
Georgia Institute of Technology

Dr. Ellen Do
School of Interactive Computing
Georgia Institute of Technology

Dr. Julie Linsey
School of Mechanical Engineering
Georgia Institute of Technology

Dr. Nancy Nersessian
School of Interactive Computing
Georgia Institute of Technology

Dr. Marc Weissburg
School of Biology
Georgia Institute of Technology

Date Approved: October 11, 2013

ACKNOWLEDGEMENTS

There are many individuals over the years who have influenced and inspired the work in this dissertation, none more so than my advisor, the generous and eternally patient Dr. Ashok Goel. He fostered an environment of exploration and argumentation, providing the creative freedom for me to grow and the critical insight and guidance to raise the quality of my ideas to the highest scientific standards.

I would like to thank each of my committee members for adding their own insight to this work: Dr. Ellen Do who provided me with practical insight into the processes of creativity; Dr. Nancy Nersessian who helped me understand the qualitative side of scientific investigation, as well of the sociology of science; Dr. Julie Linsey who provided the pragmatic perspective of a mechanical engineer, and helped iron out the wrinkles in my statistics; and Dr. Marc Weissburg who provided critical insight into my scientific reasoning, as well as contributing his experience as a biologically inspired design practitioner and instructor.

I owe tremendous gratitude to the Center for Biologically Inspired Design at Georgia Tech. I would especially like to thank the director of CBID, Dr. Jeannette Yen, who welcomed me and my theories into her classroom, with enthusiasm, energy and unwavering support. The Center for Biologically Inspired Design also provided unique opportunities to teach biologically inspired design at NASA and at Tohoku University in Japan, memories which I will always cherish.

I will forever be thankful for the experience of working with my fellow PhD students and friends at the Design and Intelligence Lab: the everyday discussions with the brilliant Swaroop Vattam; the daily dose of wit and cleverness from Bryan Wiltgen; the late night

philosophizing with Josh Jones; my study (and nacho) buddy Maithilee Kunda; the always ambitious David Joyner; and the fractal master, Keith McGreggor. I would also like to thank Marika Shahid for the many long hours of qualitative coding essential to this work.

I would like to thank my friends and family for all their support as I chased this crazy dream. My parents, Ron and Irene Helms, who helped me appreciate nature, the value of doing what you love, and who persistently nudged me in the right direction; my sisters Marie, Kristen and Kathy, who all supported their maverick brother in their own ways; my friends Steve Guenther and Richard Guenther with whom my imagination soared, and Timothy Moll, with stories that will always make me laugh; and Laura Warner, who provided loving, patient support through all the long days of writing.

Finally I would like to thank the US National Science Foundation for their financial support over the years, through NSF grants #0534622, #0632519, #0855916, and #1022778.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS	iii
LIST OF TABLES	xi
LIST OF FIGURES	xv
LIST OF ABBREVIATIONS	xix
SUMMARY	xx
1 INTRODUCTION	1
1.1 Background	1
1.1.1 Observational Studies of Biologically Inspired Design	2
1.1.2 Exploratory Experiments on Problem-Solution Interaction	5
1.1.3 Summary of Exploratory Findings	8
1.2 Research Problems and Questions	9
1.2.1 Design Problem Formulation in BID	9
1.2.2 Research Problem 1	11
1.2.3 Problem Evolution in BID	15
1.2.4 Research Problem 2	15
1.2.5 Support Tools for BID	25
1.2.6 Research problem 3	26
1.2.7 Research problem 4	30
1.3 Personal Motivation	33
1.3.1 My observations of BID class	33
1.3.2 Parallels with PhD research	34
1.4 Limitations and Assumptions	35
1.5 Delimitations	36
1.6 Organization	37
2 RELATED RESEARCH	39
2.1 Problem formulation in design	39
2.1.1 Defining design as an ill-structured problem	39
2.1.2 Two schools of thought on problem structure	40
2.1.3 Top-down design	41

2.1.4	Solution-orientation	42
2.1.5	Problem Decomposition in Design	48
2.1.6	Time Spent on Problem Structuring	49
2.2	Content accounts of problem representations	50
2.2.1	Functional representations of design problems	50
2.2.2	Design problem formulation in practice.	52
2.2.3	Analytic representations of problems	53
2.3	Design process accounts specific to biologically inspired design	54
2.3.1	Problem-driven and Solution-based design.	54
2.3.2	Solution-based Biologically Inspired Design Process	55
2.3.3	Compound Analogy	55
2.4	Applications in support of biologically inspired design	57
2.4.1	Process and Cognitive Support for Biologically Inspired Design	58
2.4.2	Support Technologies for Biologically Inspired Design	58
3	CONTEXT AND METHODOLOGY	62
3.1	Biologically Inspired Design	62
3.2	The Biologically Inspired Design Class	66
3.2.1	Course Content and Assignments	67
3.2.2	Course Instructors	69
3.2.3	Student Composition	71
3.2.4	Biologically Inspired Class Design Example: RaPower, 2009	73
3.2.5	Key Changes in the Classroom Environment 2006-2012	76
3.2.6	Role of the Researcher	81
3.3	Data and Data Collection	83
3.3.1	Student Demographics	83
3.3.2	Classroom Content	84
3.3.3	Student Assignments	85
3.3.4	Student Design Projects	93
3.3.5	Student Surveys	96
3.3.6	Student Interviews & Design Sessions	96
3.3.7	SR.BID for the Web	97
3.4	Research Method	97
3.4.1	Exploratory Phase	99

3.4.2	Content Account Development Phase _____	102
3.4.3	Process Account Development Phase _____	104
3.4.4	Tools and Interventions _____	105
4	EXPLORATORY STUDIES IN PROBLEM EVOLUTION _____	110
4.1	Study E.1: Observational Studies of the BID Class _____	111
4.1.1	Study Motivation _____	111
4.1.2	Methodology of Study _____	112
4.1.3	Results of Study _____	113
4.1.4	Summary of 2006 Observational Studies _____	117
4.2	Study E.2: Biological Source Influence on Problem Decomposition _____	119
4.2.1	Study Motivation _____	119
4.2.2	Methodology of Study _____	120
4.2.3	Results of Study _____	126
4.3	Study E.3: Qualitative Analysis of Problem-Solution Decompositions _____	128
4.3.1	Study Motivation _____	128
4.3.2	Methodology of Study _____	128
4.3.3	Results of Study _____	131
4.4	Summary of Exploratory Studies in Problem Evolution _____	133
4.5	Claim 1 _____	134
4.6	Research Problem 1 _____	134
4.7	Research Problem 2 _____	135
5	SR.BID: A CONTENT ACCOUNT OF DESIGN PROBLEMS _____	137
5.1	Existing content accounts of problem formulation in design. _____	137
5.1.1	Motivation _____	137
5.1.2	Evaluation of Theories _____	138
5.1.3	Results of the Evaluation _____	140
5.2	The Use of SBF in the Biologically Inspired Design Classroom _____	141
5.3	Study C.1: SR.BID Schema Development _____	144
5.3.2	Phase 1: Initial Coding of SR.BID _____	149
5.3.3	Phase 2: Refinement of SR.BID _____	155
5.3.4	Phase 3: Analytical Application of SR.BID _____	158
5.4	Summary of SR.BID _____	164
5.5	Claim 2 _____	165

6	A PROCESS ACCOUNT OF PROBLEM EVOLUTION _____	166
6.1	Research Problem 2 _____	166
6.2	Study P.1: Qualitative Analysis of APE using SR.BID _____	167
6.2.1	Study Motivation _____	167
6.2.2	Methodology _____	167
6.2.3	Results of the Study _____	178
6.2.4	Summary of Study _____	183
6.3	Study P.2: Developing a Process Account of APE _____	184
6.3.1	Study Motivation _____	184
6.3.2	Methodology _____	184
6.3.3	Summary of Study P.2 _____	207
6.4	Claim 3 _____	208
7	THE FOUR-BOX METHOD OF PROBLEM FORMULATION _____	209
7.1	Problem Formulation Challenges _____	210
7.1.1	Error 1. Vaguely defined problems. _____	210
7.1.2	Error 2. Poor problem-solution pairing. _____	210
7.1.3	Error 3. Misapplied analogy. _____	210
7.1.4	Finding 1: Designers struggle with design problem formulation _____	211
7.1.5	Finding 3: Designers have difficulty making “correct” analogies _____	211
7.2	Research Problem 3 _____	211
7.3	Study I.1: Accuracy of Student Use of the Four-Box Method _____	212
7.3.1	Study Motivation _____	212
7.3.2	Methodology _____	214
7.3.3	Results of the Study _____	223
7.3.4	Summary of the Study _____	229
7.4	Study I.2. Qualitative Analysis of the Four-Box Method _____	231
7.4.1	Study Motivation _____	231
7.4.2	Methodology _____	231
7.4.3	Results of the Study _____	241
7.4.4	Summary of the Study _____	250
7.5	Claim 4 _____	251
7.6	Study I.3: A Web-based Implementation of SR.BID _____	251
7.6.1	Research problem 4 _____	252

7.6.2	Study Motivation _____	252
7.6.3	The Functionality of the SR.BID Web Application _____	253
7.6.4	Study Methodology _____	269
7.6.5	Results of Study _____	274
7.6.6	Summary of Study _____	275
7.7	Claim 5 _____	276
7.8	Future Research for the SR.BID Web Application _____	276
8	CONCLUSIONS AND SUMMARY _____	278
8.1	Claims _____	279
8.1.1	Exploratory Studies _____	279
8.1.2	Claim 1 _____	282
8.1.3	A Content Account of Analogical Problem Evolution _____	283
8.1.4	Claim 2 _____	285
8.1.5	A Process Account of Problem Evolution _____	289
8.1.6	Claim 3 _____	298
8.1.7	The Four-box Method of Problem Formulation _____	299
8.1.8	Claim 4 _____	304
8.1.9	The SR.BID Web Applications _____	304
8.1.10	Claim 5 _____	308
8.1.11	Generalization of Claims _____	308
8.2	Contributions _____	312
8.2.1	Cognitive Science _____	312
8.2.2	Artificial Intelligence _____	314
8.2.3	Design Science _____	316
8.2.4	Biologically Inspired Design _____	317
8.2.5	Biologically Inspired Design Pedagogy _____	318
8.3	Future Research _____	319
8.3.1	Design studies of analogical problem evolution _____	319
8.3.2	Problem definition through design lifecycle _____	321
8.3.3	Expert versus novice studies _____	321
8.3.4	Extended studies of the four-box and T-chart methods _____	321
8.3.5	T-chart as a tool for understanding analogies _____	322
8.3.6	Extension of the SR.BID Web Application _____	323

8.4 Closing thoughts _____	324
APPENDIX A:Projected Growth of Biologically Inspired U.S. Patents _____	327
APPENDIX B:Examples of Compound Analogy _____	330
APPENDIX C:Sample Problem Decompositions _____	334
APPENDIX D:Biological System Information, Study E2 _____	340
APPENDIX E:SR.BID Sub-category Descriptions _____	349
APPENDIX F:SR.BID Coding Rubric _____	353
APPENDIX G:Example design document (2009) and SR.BID coding _____	374
APPENDIX H:Definition of Transformations _____	379
APPENDIX I: 2012 End of Semester Survey _____	390
APPENDIX J: Evaluation of open-ended 2012 survey results _____	394
APPENDIX K:ERR Diagram for the SR.BID Web Application _____	398
APPENDIX L:T-Chart Case Study _____	400
REFERENCES _____	411

LIST OF TABLES

Table 1-1. Conceptual categories, definitions and the percentage of their occurrence in student functional decomposition assignments, measured over all occurrences.	8
Table 1-2. Amount of support for a cognitive theory of analogical problem evolution, measured in terms of full support, partial support or no support, for each of six variables provided by each category of problem formulations	12
Table 1-3. The five characteristics of complex problems and the goals associated with addressing them in complex problem solving.	20
Table 1-4. Design problem strategies associated with the three design problem goals....	21
Table 1-5. The set of transformations used to change problem formulations in BID.	22
Table 1-6. Positive and negative comments associated with the 4-box method of problem formulation. Comments are gathered from student reflections at the end of the 2012 BID class, and are summarized by category.....	30
Table 1-7. Sample of transaction reports for a single user for the SR.BID Web Application.....	32
Table 3-1. Partial list of professors and lecturers for the biologically inspired design class at Georgia Institute of Technology, 2006-2012.....	71
Table 3-2. Exploratory studies and associated study type.	100
Table 3-3. Content account studies and associated study type.	104
Table 3-4. Process account summaries and associated study types.....	105
Table 3-5. Intervention studies and associated study types.	108
Table 3-6. Summary of research problems and associated studies.....	109
Table 4-1. Functional concepts added to the second decomposition and the biological organism associated with it.....	126
Table 4-2. Conceptual categories, definitions and the percentage of their occurrence in student functional decomposition assignments, measured over all occurrences.	131
Table 4-3. Decomposition types, definition of type and number of assignments classified as each type.	132
Table 5-1. Design problem formulation theory categories, and the authors associated with them.....	140

Table 5-2. Amount of support for a cognitive theory of analogical problem evolution, measured in terms of full support, partial support or no support, for each of six variables provided by each category of problem formulations	141
Table 5-3. Conceptual “seeds” of the SBF ontology	149
Table 5-4. Non-weighted mean percentage, standard deviation and frequency of occurrence of each category.....	160
Table 5-5. Percentage breakdown by core concept and concept relationship	160
Table 6-1. Example differential analysis of concepts between problem models.....	171
Table 6-2 Example differential analysis of relationships between problem models.	172
Table 6-3. Example SR.BID encoding of two paragraphs of text from a design problem definition.	174
Table 6-4, Number of concepts, by concept type across four points in time.....	178
Table 6-5. Function concept carry through from problem descriptions.	179
Table 6-6. Operational environment concepts by reference to source.....	181
Table 6-7. Function concepts by reference to source.	181
Table 6-8. Performance criteria by reference to source.	182
Table 6-9. The five characteristics of complex problems and associated goals.	190
Table 6-10. Transformation resultant states.....	190
Table 6-11. Design examples associated with complex problem characteristics.	191
Table 6-12. Design behaviors associated with complex problem solving.....	193
Table 6-13. Hypothesized design problem strategies associated with designer problem goals.	194
Table 6-14. Design problem strategy details.	196
Table 6-15. The hypothesized set of problem transformation primitives.	198
Table 6-16. Problem-solution model relationships.	206
Table 7-1. ANOVA analysis of accuracy using major and mode.....	228
Table 7-2. ANOVA analysis of accuracy using mode and year.	228

Table 7-3. ANOVA analysis of accuracy using year and major.....	229
Table 7-4. Count of self-reported majors, 2012 student surveys.....	237
Table 7-5. Sample answers to first survey question.	238
Table 7-6 Number of phrases or points for each question.....	238
Table 7-7. Survey question 2, important skills	242
Table 7-8. Survey question 3, problem concepts.....	243
Table 7-9. Survey question 4, SR.BID use	244
Table 7-10. Comments, count by category for four-box method.	246
Table 7-11. Comments, count by category for the SR.BID representation.	248
Table 7-12. Comments, count by category for the T-chart method.....	249
Table 7-13. SR.BID Web Application transaction data, User #6.	271
Table 7-14. SR.BID Web Application transaction data, User #7.	272
Table 8-1. Non-weighted mean percentage, standard deviation and frequency of occurrence of each category.....	285
Table 8-2. Function concepts, by reference to man-made, biological or unknown sources.	289
Table 8-3. The five characteristics of complex problems and associated goals.	293
Table 8-4. Hypothesized design problem strategies associated with designer problem goals	294
Table 8-5. Partial set of problem transformation primitives.....	295
Table 8-6. Transformations required to move from problem formulation 2 to problem formulation 3.....	296
Table 8-7. Problem-solution model relationships.....	297
Table 8-8. Comments, count by category for the four-box method.....	303
Table A-1. Number of US patents containing the words “biomimetic,” “bionic,” or “biologically inspired”, issued 1985-2013.....	329
Table A-2. Re-projection values vs. original projections.	330

Table B-1. Compound analogy projects.	331
Table E-1. SR.BID primary and secondary solution types	350
Table E-2. SR.BID primary and secondary function types	350
Table E-3. SR.BID primary operating environment types	351
Table E-4. SR.BID primary and secondary constraint and specification types.....	352
Table E-5. SR.BID primary performance criteria types	353
Table E-6. SR.BID primary deficiency and benefit types	353
Table G-1. SR.BID encoding of two paragraphs of text.....	377
Table J-1. Sample survey answers and encoding.....	395
Table J-2. Example survey comments and count by category.....	396
Table J-3. Critical BID skills and number of times cited in survey answers.....	397
Table J-4. Useful concepts for problem specification.	398
Table J-5. Opportunities for improving SR.BID.	398
Table L-1. Initial iteration of T-chart comparison between tire-traction problem and polar bear paws.	402
Table L-2. Second iteration of T-chart comparison between shoe-traction problem and polar bear paws	404
Table L-3. T-chart comparison between the light pole problem and bamboo.....	405
Table L-4. T-chart comparison of the light pole problem and the glass sponge	406
Table L-5. T-chart comparison of the light pole problem and the peregrine falcon.....	408

LIST OF FIGURES

Figure 1-1. The final problem decomposition of a filtration design problem created during an in-class exercise. Green boxes represent the initial (given) decomposition, blue represent the decomposition after a single iteration, pink represent the decomposition after students were provided with biological analogue systems.	6
Figure 1-2. The conceptual categories of SR.BID and their relationships.	14
Figure 1-3: Graphical representation of the PE.BID theory of problem evolution in biologically inspired design.	19
Figure 1-4 Student designers use the four-box method of problem formulation to articulate a problem in terms of (1) operational environment, (b) functions, (c) specifications/constraints and (d) performance criteria.	27
Figure 1-5: In the T-chart method of analogical evaluation, designers compare a design problem to a biological system across the four-dimensions of the four-box method, using a simple comparison: different, similar, or same.	29
Figure 2-1. Problem-solution evolution, Maher, Poon and Boulanger (1993)	45
Figure 2-2. Problem-solution evolution, Dorst & Cross (2001)	46
Figure 2-3. Analogical problem evolution.....	47
Figure 3-1 Final design diagram of SolShield from the RaPower design team.....	73
Figure 3-2. Sample syllabus from 2012 biologically inspired design class.....	85
Figure 3-3. The four-box method of design problem formulation.....	88
Figure 3-4. The T-chart method of analogical evaluation. Designers compare a design problem to a biological system across the four-dimensions of the four-box method, using a simple comparison: different, similar, or same.	92
Figure 4-1 Figure depicting particle size range for filtration design problem, included with problem description in study E.1	122
Figure 4-2. Initial “shallow” functional decomposition provided to students for the air filtration problem.	123
Figure 4-3 Problem decomposition after initial facilitated group decomposition exercise was complete.....	125
Figure 4-4. The final problem decomposition of a filtration design problem created during an in-class exercise. Green boxes represent the initial (given) decomposition, blue	

represent the decomposition after a single iteration, pink represent the decomposition after students were provided with biological analogue systems.	127
Figure 5-1. Percentage breakdown of 570 concepts by concept type.....	153
Figure 5-2: The primary conceptual categories and relationships of the SR.BID content account for problem formulation in BID.	157
Figure 5-3. Sample SR.BID problem model.....	162
Figure 6-1. Two partial models of SolShield design problem from adjacent time steps.	170
Figure 6-2: Graphical representation of the PE.BID theory of problem evolution in biologically inspired design.	187
Figure 7-1. Four box method of problem definition.	217
Figure 7-2. Student sample of four-box design problem description.	221
Figure 7-3. Histogram of accuracy of use of four-box method.	224
Figure 7-4 Differences in accuracy of 4-box method, by category, by year.	225
Figure 7-5. Difference in accuracy of 4-box method, by major.	226
Figure 7-6. Accuracy of use of four-box method, by mode of representation.....	227
Figure 7-7. Web application high-level menu.	254
Figure 7-8. Member list of existing projects.....	254
Figure 7-9. Project options.....	254
Figure 7-10. Project document entry screen.	255
Figure 7-11. Project image entry screen.	256
Figure 7-12. Partial screen shot of four-box entry screen.....	257
Figure 7-13. Screen capture from adding a new operational environment.	258
Figure 7-14. Screen shot of sub-type selection for operational environment.	259
Figure 7-15 Biological analogue source descriptions, Camel Nose.	260
Figure 7-16. Biological analogue four-box description, Camel Nose.	261
Figure 7-17. Analogy rationale, Camel Nose.	262

Figure 7-18. Screen shot of the four-box model for a biological system, Scorpion.	263
Figure 7-19. Biological system query construction.	263
Figure 7-20. Screen shot of query type drop down menu.	264
Figure 7-22. Compound “OR” query, using multiple terms and types.	265
Figure 7-21. Drop down pick list for functions containing the query term “water.”	265
Figure 7-23. Biological system queries return annotated images.	266
Figure 7-24. Description associated with a biological system, Thorny Devil.	267
Figure 7-25 Four-box model of biological system, Thorny Devil.	268
Figure 7-26 Adding an existing biological system to a design project.	269
Figure 8-1. The final problem decomposition of a filtration design problem created during an in-class exercise. Green boxes represent the initial (given) decomposition, blue represent the decomposition after a single iteration, pink represent the decomposition after students were provided with biological analogue systems.	281
Figure 8-2. The primary conceptual categories and relationships of the SR.BID content account.	287
Figure 8-3. Sample SR.BID problem model.	288
Figure 8-4. Graphical representation of the PE.BID theory of problem evolution in biologically inspired design.	292
Figure 8-5. Partial problem model from problem formulation 2.	296
Figure 8-6. Partial problem model from problem formulation 3.	296
Figure 8-7. Histogram of accuracy of use of four-box method.	300
Figure 8-8. Differences in accuracy of 4-box method, by category, by year.	301
Figure 8-9. Partial screen shot of four-box entry screen.	306
Figure A-1: Cumulative total of US issued patents 1985-2012.	329
Figure A-3. Re-projection of BID patent growth, using US patent data through 2012. .	330
Figure B-1. Generation of compound analogy for Eye in the Sea.	332
Figure B-2. Generation of compound analogy for InvisiBoard.	333

Figure C-1. Functional decomposition for movement over sand by instructors.....	335
Figure C-2. Functional decomposition for movement over sand, by students	336
Figure C-3. Functional decomposition for stealthy undersea movement and copepod..	337
Figure C-4. Functional decomposition of plant growth and solar energy conversion....	338
Figure C-5. Functional decomposition of plant growth and clean building.	339
Figure C-6. Functional decomposition of Atlanta traffic model and ant traffic mode, showing constraint mismatch.....	340
Figure D-1. Electron micrographs of the shoulder region of cirrus	341
Figure D-2. Particulate capture	342
Figure D-3. Bowhead Whale	343
Figure D-4. Filtration process of a bowhead	343
Figure D-5 Cyclosalpas/Larvacean.....	346
Figure D-6. Hemoglobin.....	348
FigureD-7. Cooperative binding.....	348
Figure H-1. Conceptual refinement transformation.....	381
Figure H-2. Associative addition transformation.....	382
Figure H-3. Shifting-up transformation.	383
Figure H-4. Shifting-down transformation	384
Figure H-5. Induced abstraction transformation.....	385
Figure H-6. OR-type decomposition transformation.....	386
Figure H-7. AND-type decomposition transformation.....	386
Figure K-1. Entity relationship diagram of problem-solution memory model.....	400

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
APE	Analogical Problem Evolution
BID	Biologically Inspired Design
BID@GATech	Biologically Inspired Design at Georgia Tech
BIO	Biology
BME	Bio-Medical Engineering
CBID	Center for Biologically Inspired Design
DPG	Designer Problem Goal
DPS	Designer Problem Strategy
ISYE	Information Systems Engineering
FR	Functional Representation
ME	Mechanical Engineering
PD	Problem Description
PE.BID	Problem Evolution for Biologically Inspired Design
PTFE	Polymer Textiles and Fiber Engineering
SBF	Structure-Behavior-Function
SR.BID	Structured Representation for Biologically Inspired Design
WWH	What-Why-How

SUMMARY

Biologically inspired design (BID) is a widespread and growing movement in modern design, pulled in part by the need for environmentally sustainable design and pushed partly by rapid advances in biology and the desire for creativity and innovation in design. Yet, our current understanding of cognition in BID is limited and at present there are few computational methods or tools available for supporting its practice. In this dissertation, I develop a cognitive model of BID, build computational methods and tools for supporting its practice, and describe results from deploying the methods and the tools in a Georgia Tech BID class.

One key and novel finding in my cognitive study of BID is the surprisingly large degree to which biological analogues influence problem formulation and understanding in addition to generation of design solutions. I call the process by which a biological analogue influences the evolution of the problem formulation *analogical problem evolution*. I use the method of grounded theory to develop a knowledge schema called SR.BID (for structured representations for biologically inspired design) for representing design problem formulations. I show through case study analysis that SR.BID provides a useful analytic framework for understanding the two-way interaction between problems and solutions.

I then develop two tools based on the SR.BID schema to scaffold the processes of problem formulation and analogue evaluation in BID. I deployed the two tools, the four-box method of problem specification and the T-chart method of analogical evaluation, in a Georgia Tech BID class. I show that with minimal training, the four-box method was used by students to complete design problem specifications in 2011 and 2012 with 75%

of students achieving better than 80% accuracy. Finally I describe a web-based application for interactively supporting BID practice including problem formulation and analogue evaluation.

Thus, my dissertation develops a cognitive model of analogical problem evolution in BID, a knowledge schema for representing problem formulations, a computational technique for evaluating biological analogues, and an interactive web-based tool for supporting BID practice. Through a better cognitive understanding of BID and computational methods and tools for supporting its practice, it also contributes to computational creativity.

1 INTRODUCTION

“When all you have is a hammer, every problem looks like a nail.” (Maslow, 1966). In the context of design, perception of the design problems changes depending on the available solutions. But how and why do design problems change, and what role does the hammer play? Moreover, in the context of innovative design – a major global economic driver – what is the role of problem inception and evolution; and again, what does that hammer have to do with it? Using the context of biologically inspired design (BID) as a domain of investigation, considering biological solutions (in lieu of hammers), I will endeavor to answer these questions and more.

1.1 Background

Biologically inspired design (BID), also known as biomimicry, biomimetics, or bionics, motivated by the need for innovation and driven by a heightened cultural awareness and desire for sustainable design, is a rising method of design. BID espouses leveraging naturally evolved systems and the discoveries made in a 3.8 billion year old design laboratory in which only the best designs survive. As a domain for innovation, BID is associated with at least 3,500 new US patents, a number which is projected to double in the next 5 years.¹

In addition to generating innovative designs, BID provides novel opportunities for the study of analogical design in design practice. The practice of BID relies fundamentally on the process of analogical design; the transfer from the domain of biology to the domain of

¹ Based on an extension of the study of Bonser 2007, see Appendix A.

engineering². Whereas much research in analogy explored the processes of analogical design computationally, in the lab, and through historical accounts, BID now provides an active, growing *in situ* environment in which to observe of analogical design in practice. Furthermore, since BID incorporates a less well explored domain for design theories, that of biology, it also provides a new domain in which to further develop and extend existing theories of analogical design.

BID also provides a unique opportunity for the study of human-computer interaction in both design and in pedagogical practice. The practice of BID dates back to at least Da Vinci, and is likely far older. However, the systemization of BID as a formal design method is a much more recent endeavor. Because the field of biologically inspired design is nascent, the processes and products developed by the community of practice are neither fully understood nor have prescriptive methods taken deep root. This provides a unique opportunity for the study of new tools and technologies in community relative free from incumbent processes and methods. In this dissertation, I will build and apply cognitive models of BID and deploy tools and processes to this community, changing at least the local landscape of BID practice.

1.1.1 Observational Studies of Biologically Inspired Design

In the context of a series of exploratory studies in 2006 and 2007 in an interdisciplinary BID class at Georgia Institute of Technology, I made three findings that are important for the future development of the discipline of BID.

² I will use engineering as the typical application of BID, although it is not limited to engineering; alternatively, architecture, computer science, or one of many other design fields may be substituted. I will specify when the domain of discussion is limited to engineering only.

1.1.1.1 *Finding 1: Designers struggle with design problem formulation in BID*

I found that in a design context that stresses innovation and creativity, where designers are allowed to determine their own design problems, student designers struggle to formulate their design problem. I observed that student design teams formulate and evolve (incrementally reformulate) their design problem, often with radical transformations. This struggle is ongoing, and often dramatic. In one observed design project, over the course of the project (one semester) the design team was observed to discard 87% of the problem-related function concepts discussed throughout the design; and only 8% of problem-related function concepts *initially* discussed were present in the *final* design. While high conceptual turnover allows for broad exploration, it comes at the expense of deep understanding of the design problem, which in turn leads to naively conceived design solutions. In the observed context, there was no explicit support – lectures, assignments, references, or tools – for design problem formulation.

1.1.1.2 *Finding2: Design problem formulation evolves in response to biological analogies.*

I found that design problems evolve in response to analogical sources from distant domains. I refer to this phenomenon as analogical problem evolution (APE). A design problem may provoke consideration of an analogy, which then instigates an alteration to the design problem formulation. This new design problem formulation may in turn generate new criteria for retrieving and evaluating additional analogies, which may in turn alter the design problem formulation, and so on. Three observations support of this finding. First I observed that some design processes are solution-based designs, that is, the design problem is defined in terms of an already-identified solution. Second, I

observed the phenomenon of compound analogical design in which multiple biological analogues were used during an extended design episode. In compound analogical design it was found that a biological analogy can initiate a decomposition of the problem in a way that the design team had not yet considered. For example, in one observed case, the design team upon learning of a biological analog with both slow- and fast-moving modes of stealthy movement decomposed their problem of stealthy movement into slow and fast modes. Third, I observed that concepts associated with biological analogues that were considered during the design process, such as a particular function or environmental condition, were perpetuated throughout a design, even though the biological analogue was no longer discussed nor used to generate intermediate or final solutions. While existing *theories of analogical design* account for the observed solution generation aspects of BID, these theories do not fully account for the problem evolution aspect of the APE phenomenon. Likewise, existing *theories of design problem evolution* do not account for the influence of analogous solutions.

1.1.1.3 *Finding 3: Designs have difficulty finding and making “correct” analogies*

I found that difficulty in defining the design problem translates into difficulties in making analogies. This is not unexpected since analogical theories stipulate a “target problem” which forms the basis for many processes of analogy, from retrieval to mapping to transfer to storage³. In my initial studies I observed that students both had difficulty (a) finding appropriate analogies, and (b) applying the analogy correctly to their design problem, both of which would result from a poorly defined problem. As an example of an incorrectly applied analogy, a design team applied a “round-trip” ant-

³See Gentner (1983, 1989), Falkenhainer, Forbus, & Gentner (1989), Holyoak and Thagard (1989),

based resource gathering model to a “one-way” traffic-control problem. The design team did not recognize that their problem was framed as a one-way problem while the solution was framed as a round-trip, and as a result they did not properly adapt the model to solve their problem. The challenge of an imprecise or dynamic problem-target is not unique to BID; for example in scientific inquiry, problem formulation is likewise dynamic. Nersessian & Chandrasakaran (2009) provide a description of the use of analogy in such a context. Moreover that problem definitions change over time in design is well known. Thus while this observation appears intrinsic to BID, it generalizes to any design domain where analogies may be found.

1.1.2 Exploratory Experiments on Problem-Solution Interaction

In 2007 I conducted two exploratory experiments to better understand the nature of the interaction between problem definition and biological analogues. In the first experiment, informed by my work on compound analogy in which students used analogues to facilitate problem decomposition (Helms, Vattam & Goel, 2008), I sought to better understand the role of biological analogies in problem decomposition.

1.1.2.1 Research Question E.1

To what extent do biological systems influence functional decomposition of problems?

1.1.2.2 Hypothesis E.1

The introduction of biological analogues to student designers will yield greater range of concepts in a functional decomposition of a design problem, than a decomposition without biological analogue prompts.

1.1.2.3 Method E.1

In this experiment a facilitated functional decomposition was carried out for a single problem as a group exercise in class, until students were satisfied with the decomposition. Student groups were then provided with different sources of biological systems with a diverse range of functions. The students were then asked to collectively further decompose the design problem. Figure 1.1 shows the results of their final decomposition.

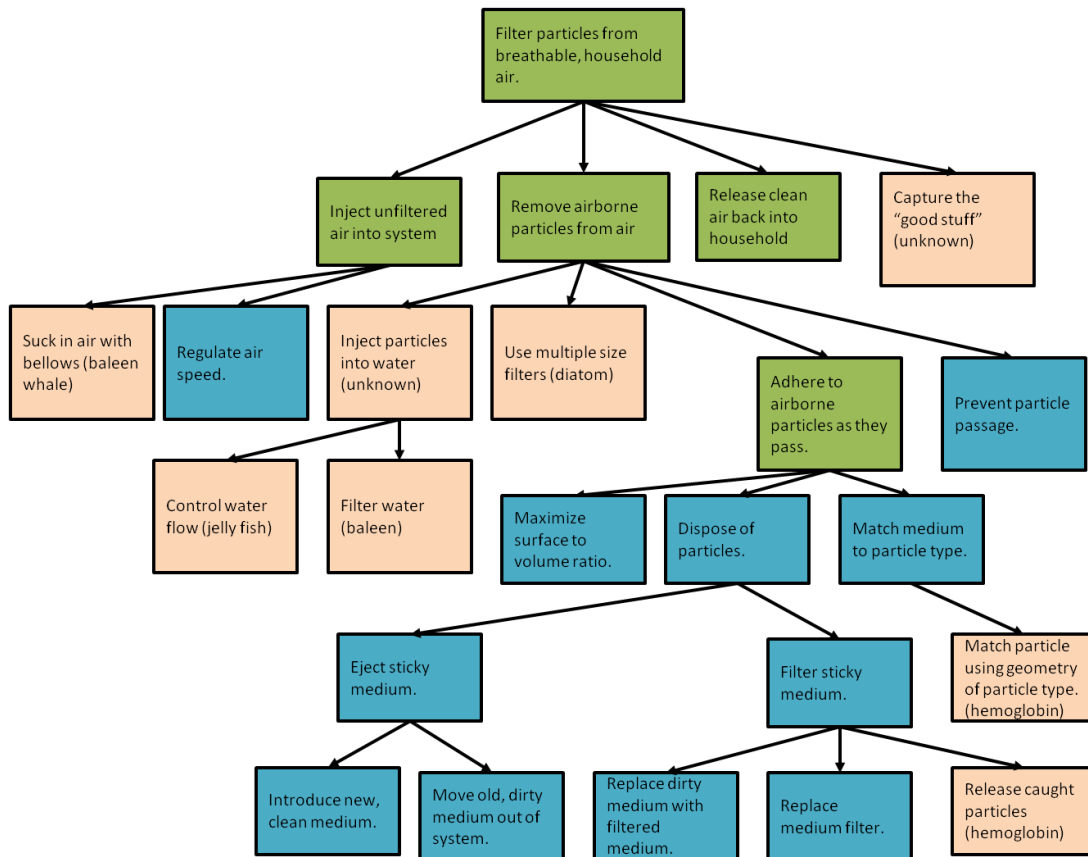


Figure 1-1. The final problem decomposition of a filtration design problem created during an in-class exercise. Green boxes represent the initial (given) decomposition, blue represent the decomposition after a single iteration, pink represent the decomposition after students were provided with biological analogue systems.

Students after exposure to the biological sources were able to add 50% more new functions than they had described in their previous functional decompositions. Functions

were added at every level of abstraction in the decomposition, and across all major branches of the decomposition. Most of these (6 of 8) additions could be traced directly back to one biological source of inspiration.

Although this study involved purely functional decompositions, I noted that students often referred to other concepts, such as structures, other solutions, and environmental factors in their decompositions. The next experiment followed up on this notion by examining the different kinds of concepts students used in “functional” decomposition assignments.

1.1.2.4 *Research Question E.2*

To what extent are student problem decompositions purely functional versus a mix of functional and other conceptual categories?

1.1.2.1 *Hypothesis E.2*

Student problem decompositions will follow a mixed conceptual decomposition strategy.

1.1.2.1 *Method E.2*

After training and several exercises in class on functional decomposition, in which both instruction and examples emphasized decompositions that were purely functional, students were asked to submit functional decompositions of problems as assignments in class. The composition of decompositions was analyzed by conceptual type. Table 1-1 provides a definition of the different conceptual types used, and the relative frequency of their appearance in “functional” decompositions.

Table 1-1 Conceptual categories, definitions and the percentage of their occurrence in student functional decomposition assignments, measured over all occurrences.

Category	Definition	Percentage Occurrence
Function	A <i>verb-noun</i> phrase A <i>verb-self</i> phrase (self implied) A <i>biological</i> function (e.g. photosynthesis)	40.57%
Function (refinement)	One or more <i>means</i> of accomplishing the function (e.g. pollination by insects, by air, by hand); One or more <i>prepositional extensions</i> of the function (e.g. movement on water, on air, on land)	5.42%
Structure	A <i>property, component, or material</i> composition of a solution (e.g. the color red, a flower petal, and protein respectively)	26.89%
External Factors	The <i>environment</i> (e.g. in the forest) or a <i>condition</i> of the environment (e.g. partially shaded) external to the system.	5.19%
Solution	<i>Solution</i> is used to perform function, <i>Solution</i> performs function itself <i>Solution</i> described a method for performing function	18.16%
Behavior (causal)	A simple causal phrase (A causes B) A complex causal description	3.77%

1.1.3 Summary of Exploratory Findings

The process of solution-based design, occurring naturally in roughly half of the observed cases of BID, depends on an initial seed biological source from which a principle may be extracted and which in turn prompts problem inception. Compound analogy, occurring equally as frequently, entailed the use of multiple analogues in the development of a solution to a system. A compound analogy is often the result of a partitioning of a design problem into independent sub-problems each of which can be addressed by a different biological source. The cause for this partitioning is often a biological source itself, as in the stealthy, but low-speed copepod in the example.

The experiment in problem decomposition demonstrated that when student designers are prompted with biological analogues, they are capable of redefining a design problem

at almost any level of abstraction. Finally, I show that in solution-based problem decompositions solution-dependent concepts such as the parts or materials of a biological system, serve as fundamental conceptual components of student problem formulation, occurring equally as frequently as functional concepts. Taken in combination, this evidence demonstrates that when student designers formulate problems in the biologically inspired design classroom context, beginning with problem inception and continuing throughout conceptual design, biological analogues influence problem conceptualization.

1.2 Research Problems and Questions

Supported by my observational and exploratory studies, the initial research problems concern the development of an underlying theory of analogical problem evolution (APE) in BID, to be followed by interventions based on those theories. One productive means to frame a theory of analogical design is to ask four questions: why, what, how and when (Goel, 1997). In this framing, the “why” pertains to the task for which the analogy is used, the “what” pertains to the content of knowledge, the “how” pertains to the methods, and the “when” pertains to strategic process control. I will begin with the development of the “what” which I will call the content account.

1.2.1 Design Problem Formulation in BID

Student design performance suffers as a result of the large number of concepts that are dropped through the design process in design problem formulation. In each design problem reformulation, some design thinking must be cast aside or reworked to integrate into the new design problem conceptualization. While many theories of design account

for design problem reformulation as a high level process account, (Hillier et al, 1972; Darke 1979; Maher et al 1993; Dorst & Cross, 2001; Pahl & Beitz 2003, to name a few) many are silent on the content and methods of design problem reformulation. Some theories of design do specify design problem representations, and can be grouped according to the following four categories:

1. Normative, based on best practice (Wirth 1971; Dahl, Dykstra, and Hoare, 1972; Roozenberg & Eekels, 1995)
2. Normative and based on function/function decomposition (Altshuller, 1984; Sturges et al, 1996; Kirshman, Fadel & Jara-Almonte, 1998; Hundal, 1990; Stone and Wood, 2000; Pahl & Beitz 2003)
3. Abstract, computational accounts (Newell & Simon, 1972; Simon, 1973; Gero, 1996)
4. Solution-generation focused accounts (Goel and Chandrasakaran, 1989; Goel, 1992; Bhatta and Goel, 1994; Gero, 1990; Gero and Kannengiesser, 2004; Sarkar and Chakrabarti, 2008).

While any of these representations may be used to support design problem formulation, and many have, they were not conceived with the goal of supporting the task of *design problem formulation* in the context of *analogical design*, or in the context of BID specifically. BID requires support of broader processes (analogical retrieval, mapping, transfer, and evaluation) and domains (biology) than is required for traditional engineering design⁴. Additionally, many of these theories were not designed or intended

⁴ This is not to say that analogical design and/or biological sources may not occur in traditional design. Rather that they are neither typical nor required, and thus not necessarily supported as they must be in BID.

to be used in support of a *cognitive account of design problem formulation*⁵.

In developing a theory that supports analogical problem evolution in biologically inspired design, I will begin by providing a representation, or content account. I focus initially on the content account, rather than a process account, for three reasons. First, with a content account I can more accurately and consistently describe the phenomenon of design problem formulation, including how that content changes over time. Second, the content account provides the underlying language for describing the process account; that is the content account provides the set of concepts over which the process account must act. Third, much as a requirements gathering document may be used to facilitate problem definition in domains in which best practices are well established, a content account for problem formulation in BID may provide a principled method for developing tools to facilitate and focus the problem formulation and related tasks. This leads to the first research problem.

1.2.2 Research Problem 1

While many theories of design problem representation exist, it is unknown to what extent current content theories of design support analogical design problem formulation and evolution in BID.

1.2.2.1 Literature Review

I first evaluate existing design literature against a set of criteria necessary to fully support process of problem evolution in biologically inspired design. The degree to which a design theory may be considered to support a cognitive theory of design problem evolution in BID may be inferred based on six criteria: (a) the taxonomy of problem

⁵ While the abstract, computational accounts do provide insight into design cognition, they do so at a very high level e.g. providing descriptions in terms of state spaces and state space search.

concepts, (b) the taxonomy of problem concept relationships, (c) support for the biological domain, (d) support for processes of analogy, (e) support for processes of problem-evolution, and (f) support for cognitive models.

Each design theories may be categorized into one of four main types of design theory. I evaluate each category of theories with respect to my six criteria. The evaluation of each category of theory is based on evaluating the capability of *any* theory to fulfill the requirements of the variable. Each category is ranked on a three point scale: full support, partial support or none (does not support). This evaluation establishes the extent to which each theory category provides an underlying cognitive account for problem evolution in BID. Table 1-2 shows the evaluation results.

Table 1-2. Amount of support for a cognitive theory of analogical problem evolution, measured in terms of full support, partial support or no support, for each of six variables provided by each category of problem formulations

Variable	Normative	Normative-Functional	Abstract, computations	solution-generation
Categories	Full	Partial	None	Partial
Relationships	Partial	Partial	None	Partial
Biological Domain	None	Partial*	None	Full
Analogical Focus	None	Partial*	None	Full
Problem Focus	Full	Full	Full	None
Cognitive Focus	None	Partial	Full	Full

* recently developed theories and applications support some aspects of BID

The above table shows that no single category of theories supports a comprehensive content account of BID. It also shows that comparatively, the solution-generation design

theories do provide a higher level of support than others. Of these, SBF and SAPHhIRE models appear to be most promising. This leads to my second research question.

1.2.2.2 *Research Question C.1*

What adaptations to the solution-generation oriented theories are needed to fully support a content account of design problem formulation and evolution in BID?

1.2.2.3 *Hypothesis C.1*

SBF provides a partial content account of analogical design that may be used as a seed ontology to discover the underlying account of problem formulation and evolution in BID.

1.2.2.4 *Method C.1*

I use a modified form of grounded theory, called ontologically grounded theory, to show that SBF as a seed ontology may be applied to problem formulation data to form a comprehensive account of the content account. The account is enriched by adding, modifying or deleting concepts and relationships as required, through several iterations. The resulting ontology, Figure 1.1, is Structured Representations for Biologically Inspired Design, SR.BID. I apply the SR.BID content account to new data, and validate the comprehensiveness and accuracy of the model using standard measures of inter-coder and intra-coder reliability.

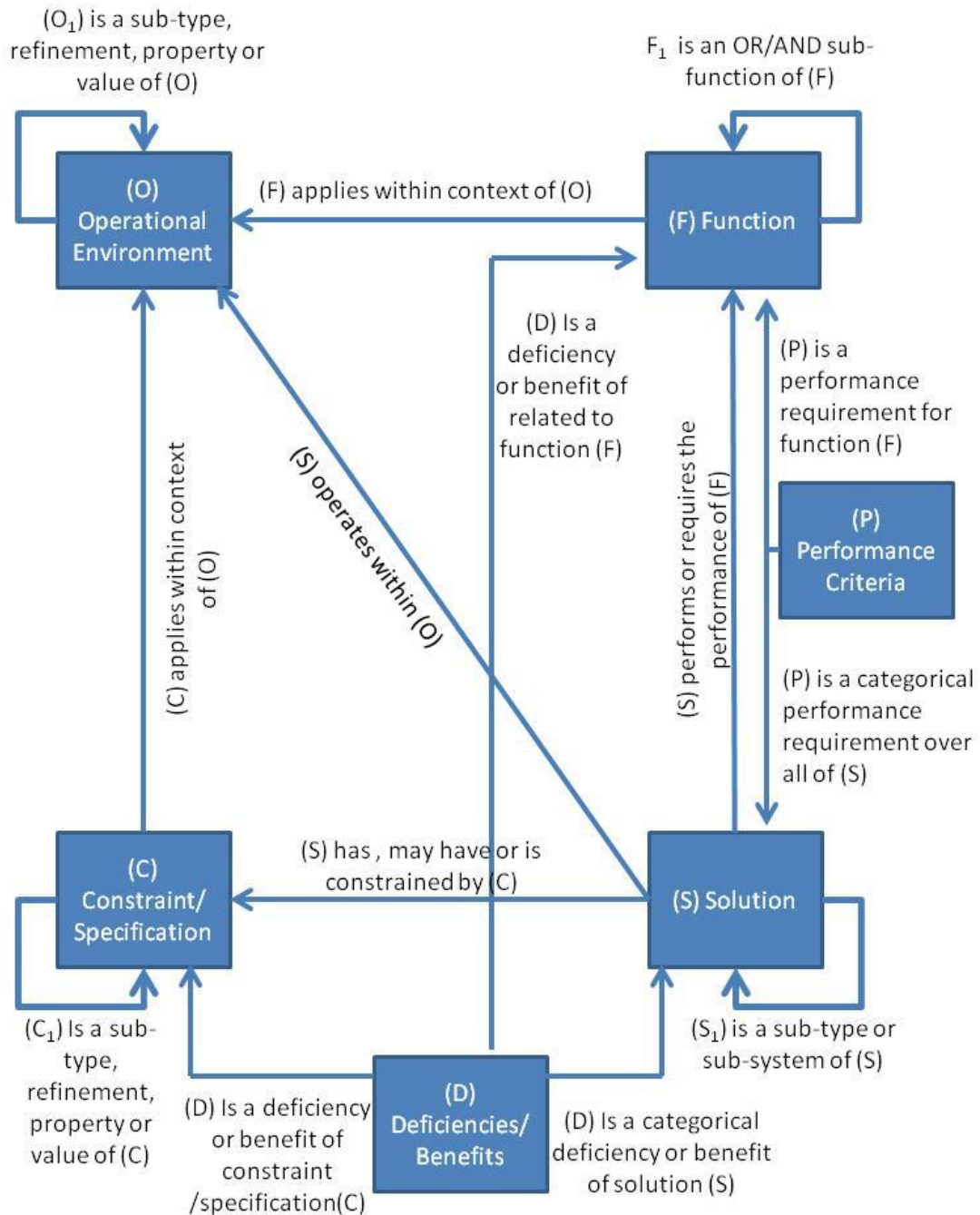


Figure 1-2. The conceptual categories of SR.BID and their relationships.

1.2.2.5 Summary of Section

In this section I establish the adequacy of existing theories of design to address

knowledge content portion of a theory of analogical problem evolution in BID. I leverage SBF to derive a new content account of problem formulation, called SR.BID. I validate SR.BID against data generated in in the BID design course.

1.2.3 Problem Evolution in BID

Analogical problem evolution (APE) as a phenomenon lies at the intersection of two not yet integrated aspects of design theory: analogical design and problem-solution coevolution. From the perspective analogical design theories, APE exhibits the classic retrieval-mapping-transfer behaviors one would expect in those theories. In traditional theories of analogical design, especially theories of analogical design as applied to BID, transfer occurs between a biological source and a conjectured design solution. APE exposes a new opportunity for the application of analogical theories of design to not only solution generation, but to problem evolution as well. Second, as an instance of problem-solution coevolution, APE describes a phenomenon whereby a design problem evolves in response to the evaluation of a solution. Unlike in traditional problem-solution coevolution, however, the solution in question is neither a new conjectured solution nor a solution within the current (engineering) domain. Rather APE uses an existing (analogue) solution from the domain of biology. APE provides the opportunity to extend existing problem-solution coevolution theories on one hand into cross-domain solutions and on the other hand into existing solutions, as a means to evolve problems.

1.2.4 Research Problem 2

Current theories of analogy account for analogical evolution of solutions; while current theories of problem evolution account for problem evolution in the context of conjectured solutions. Analogical problem evolution in response to existing

analogical solutions remains unaccounted for in design theory.

Until now, descriptions of APE have been inferred from high level task-level accounts of related phenomena, such as solution-based design and compound analogy, or from observational accounts such as the design trajectory accounts generated from my 2006 and 2007 studies. There is as yet no systematic description of **what** knowledge content is transferred in the process or when the transfer takes place. The SR.BID content account enables the systematic encoding and analysis of the underlying problem models associated with the process. I will use SR.BID content model to provide a detailed description of changes to the problem model over time, and to relate those changes to concepts to identified analogies. This will provide a richer description of **when** and **what** is transferred in analogical problem evolution, which may then be used to inform a process account of APE.

1.2.4.1 Research Question P.1

What is the content is transferred from biological analogues to problem formulations, and *when* is it transferred in APE?

1.2.4.2 Hypothesis P.1

An encoding of design problem formulations in terms of SR.BID provides a reliable method for describing the content transferred between biological solutions and design problems in the process of BID.

1.2.4.3 Method P.1

APE is a subset of problem formulation in BID. Using the SR.BID content account of problem formulation in BID, I encode data collected over the course of an extended BID design episode. The data include point in time, self-generated descriptions of problem

formulations, biological analogues, and solutions generated during the semester long course of the design. I construct from these encodings for each point in time a design problem model. Qualitative differential analysis of the models, which include biological analogues, will be used as indicators of conceptual transfer from biological analogues to problem conceptualization.

Initial analysis provides a breadth of additional information for consideration in a process model of APE:

- Either existing man-made, existing biological solutions or both are cited with respect to the formulation of the problem; this occurs in all problem formulations thus far observed. Thus design problem formulation and existing solutions appear to be deeply connected.
- Concepts from biological solutions are not used in the initial problem formulation for this design episode; rather existing man-made solutions provide the basis for the concepts in the initial formulation. Concepts from biological solutions are integrated in later stages.
- Certain conceptual categories are more common between problem formulation and biological analogues than others. For example, while 38% of functional concepts that appear in problem formulations also appear in biological analogues, only 20% of performance criteria are found in common, only 17% of operational environments, and no specifications/constraints appear in common.

In light of these insights, I follow with the conjecture of an initial, high level process account of analogical problem evolution for biologically inspired design, called the

(PE.BID) model. The PE.BID model will provide an account for **why** and **how** analogical problem evolution occurs. I will break down and investigate this conjectured process account in terms of six components.

1.2.4.4 *Research Question P.2*

What is a process theory of problem evolution in design that supports the observations made of APE in the context of BID?

1.2.4.5 *Hypothesis P.2*

The *problem evolution for BID* theory (PE.BID) of analogical problem evolution supports (a) the observations made of APE, (b) the conditions required for analogical identification, mapping, and transfer, and (c) the conditions required for design problem evolution.

1.2.4.6 *Method P.2*

I will first propose PE.BID, a process account of problem evolution. Figure 2 provides a graphical representation of the PE.BID theory. This account specifies the processes and underlying memory requirements for describing **why** and **how** problem evolution occurs. The PE.BID model provide a framework to scaffold the investigation of the model components; e.g. problems, goals, strategies, memory, and transformations. For each component, I will conjecture a hypothesis and provide a method of evaluation for that hypothesis, providing results where investigations are complete. I will restrict detailed investigation for this dissertation to the transformation component. For purposes of this introduction, I will restrict discussion to the hypothesis associated with each component, deferring details to Chapter 6.

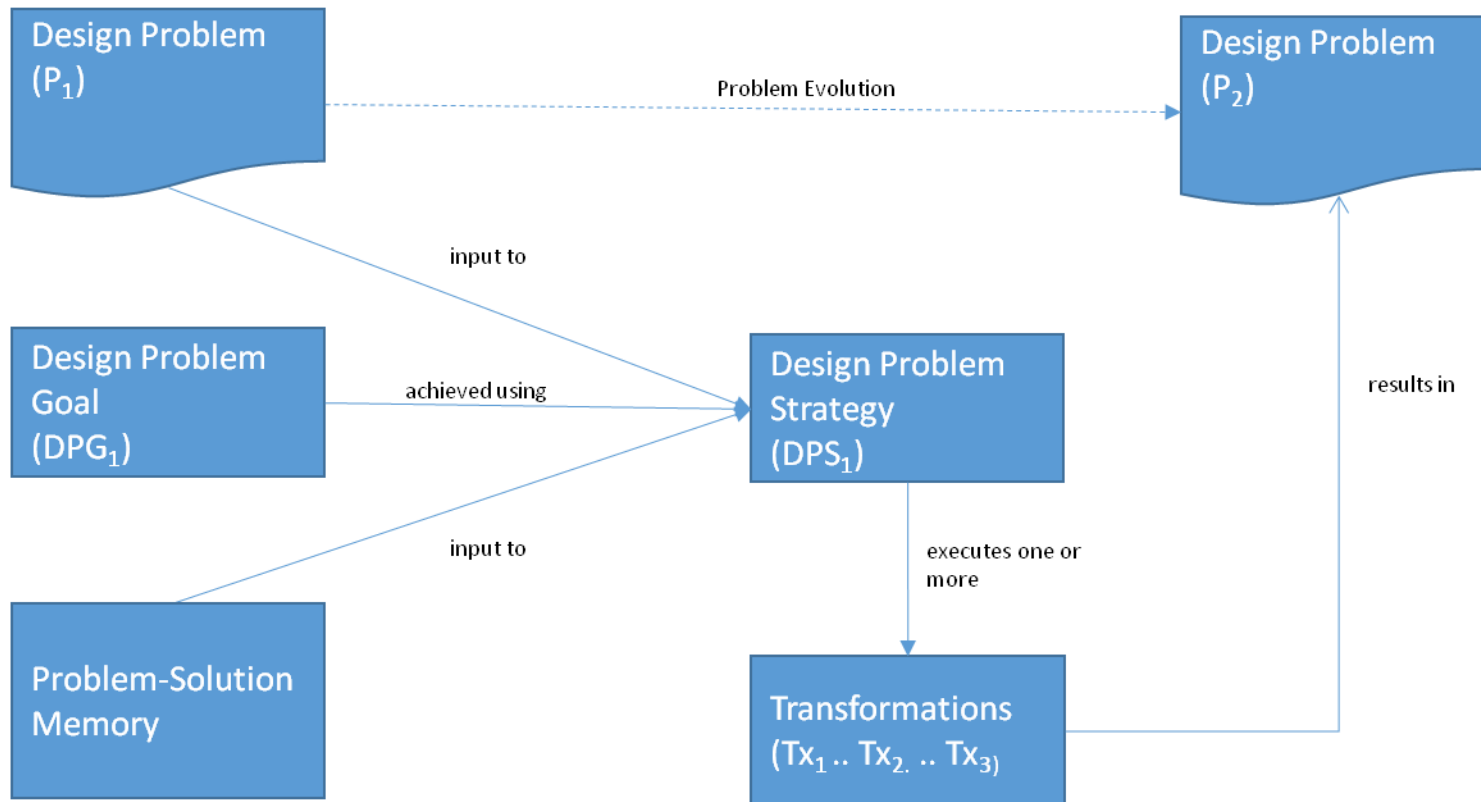


Figure 1-3: Graphical representation of the PE.BID theory of problem evolution in biologically inspired design.

1.2.4.7 Designer Problem Goals

According to Funke (2001) complex problems exhibit five characteristics that simple problems do not. I hypothesize that designers generate similar problem goals to resolve the difficulties that arise from these five characteristics.

1.2.4.8 Hypothesis P.3

The following table provides the five characteristics from Funke (2012) that ground the taxonomy of designer problem goals.

Table 1-3. The five characteristics of complex problems and the goals associated with addressing them in complex problem solving.

Characteristic	Description (Funke, 2012)	Goal
Intransparency	Intransparency concerns the variables involved and the definition of the goal. In an intransparent situation, not all required information about variables and possible goals are given.	Intransparency requires from the problem solver the active acquisition of information.
Complexity	Complexity is defined based on the number of variables (concepts) in the given system.	Complexity demands from the problem solver a simplification through reduction.
Connectivity	It is not the pure number of variables that is decisive for the workload on the problem-solving person, but the connectivity between these. Assuming that in a system of 100 variables every variable is connected to only exactly one other, the connectivity is lower than in a system in which all variables are connected to each other.	For making mutual dependencies understandable, a model of the connectivity is required from the problem solver.
Dynamics	This feature explains the fact that interventions into a complex, networked system might activate processes whose impact was possibly not intended. It signifies that in a lot of cases the problem does not wait for the problem-solving person and his/her decisions, but the situation changes itself over time.	Dynamic requires from the problem solver the consideration of the factor "time."
Polytely	Usually there is more than one goal in a complex situation that has to be considered. These goals may be in conflict.	Conflicts due to antagonistic goals require the forming of compromises and the definition of priorities.

1.2.4.9 Design Problem Strategies

Goals will in turn lead to strategies which provide the context for *why* and *when* APE is invoked, that is, APE is invoked in response to using particular strategy. I will hypothesize a small set of strategies that may be employed to achieve some design problem goals. As with goals, I will validate the hypothesis by mapping each strategy to one or more observed design examples.

1.2.4.10 Hypothesis P.4

Table 1-4 provides a small conjectured set of design problem strategies relative to the first three goals in hypothesis P.3.

Table 1-4. Design problem strategies associated with the three design problem goals.

Goal	Strategy	Description
Active acquisition of information	Breadth-first addition	Loosely related concepts are added in a breadth-first fashion, expanding the design space.
	Depth-first addition	Sub-concepts are added to existing concepts, generating conceptual depth for a particular concept.
	Relationship addition	Relationships are established between concepts
Simplification through reduction	Elimination	Concepts are removed from consideration in the problem space
	Decomposition	Concepts are divided into sub-concepts that can be considered independently; an interface may be necessary.
	Partitioning	Concepts are grouped into connected sets that can be considered independently; an interface may be necessary.
Connectivity	Relationship addition	Relationships are established between concepts
	Depth-first addition	Sub-concepts are added to existing concepts, generating conceptual depth for a particular concept.

1.2.4.11 Problem Transformations

Transformations describe the low-level operations that may occur over either problem

concepts or relationships that result in the gradual shift in the problem model over time. They are a “goal-neutral” set of primitives which may be combined to effect larger problem transformation strategies. Although I assume SR.BID as the underlying ontology, the only necessary aspect of the ontology is that of partonomic and taxonomic abstraction. The transformations may be generalized to any number of possible problem model representations.

1.2.4.12 Hypothesis P.5

Based on my own historical observations of problem formulations in the context of BID and assuming the SR.BID content account, I hypothesize the following set of primitive operators, called transformations, used to change problem formulations over time in BID.

Table 1-5. The set of transformations used to change problem formulations in BID.

Type	Sub-Type	Tertiary Type	Start State	End State
Addition	Refining		A	$A \rightarrow (r)B$
	Associating		A	$A \rightarrow (a)B$
	Abstraction shifting	Shifting-up (zooming out)	A(1-1)	$A(1) \rightarrow A(1-1)$
		Shifting-down (zooming in)	A(1)	$A(1) \rightarrow A(1-1)$
	Induced abstraction		A(1-1), A(1-2)	$A(1) \rightarrow (A(1-1), A(1-2))$

Table 1-5 continued

	Decomposing	Conjunctive	$A(1)$	$A(1) \rightarrow (A(1-1) \text{ AND } A(1-2))$
		Disjunctive	$A(1)$	$A(1) \rightarrow (A(1-1) \text{ OR } A(1-2))$
	Disconnected		--	A
Removal	Suppressing		$A \rightarrow B$	$A, \neg B$
	Deleting	Disconnected	A	$\neg A$
		Dependent chain	$A \rightarrow B$	$\neg A, \neg B$
		Partition	$(A \rightarrow B) \rightarrow C$	$\neg(A \rightarrow B), C$
	Reemerging	Related	$A \rightarrow B; \neg B$	$A \rightarrow B$
		Novel	$A \rightarrow B; \neg B$	$C \rightarrow B$
Connecting	Connect		A, B	$A \rightarrow B$
	Disconnect		$A \rightarrow B$	A, B
	Switch Connection		$A \rightarrow B$	$A \rightarrow C$
Organizing	Partitioning		A, B, C, D, E	$(A \rightarrow B) \rightarrow C \rightarrow (D \rightarrow E)$
	Decoupling		A, B, D, E	$(A \rightarrow B)(D \rightarrow E)$

1.2.4.13 Problem-solution Memory

How are the memories of design problems and design solutions organized in memory such that problem and solution concepts may be partially integrated? The final

component of the PE.BID theory is a theory of memory content and organization in which problem and solution memory is partially integrated using the SR.BID content account. First, I assume that SR.BID provides the memory organization scheme for problem formulation. Next, I note that the core concepts in the problem formulation: function, environment, performance criteria and specifications/constraints, have corresponding concepts in solution representation. However, it is not necessarily the case that all concepts in a problem representation are also represented in a solution; for example, while a problem may specify a function at a high level, the lower level functions which a particular solution implements may not be. Likewise, not all performance criteria in a problem formulation may be relevant to a particular solution. Moreover, evidence shows that all problems defined in the context of BID, are defined in terms of one or more solutions.

1.2.4.14 *Hypothesis P.6*

SR.BID provides an organization schema for a shared memory between problem formulations and solutions; solutions and problems share at least environment, function, performance and constraints/specifications in common. I have already documented some aspects of the connection between design problems and existing solutions; for example, from coded problem formulations we see that all problem models cite existing solutions. The degree to which they cite different types of concept varies by conceptual type. Additionally, one can test memory organization using a computational tool to test that SR.BID can, in principle, be used to create a memory for both biological solutions and design problems. The hypothesis can be validated computationally by demonstrating that such a memory can be instantiated and used in an application for the tasks of retrieval,

mapping and transfer.

1.2.4.15 *Summary of section*

Starting from the phenomenon of analogical problem evolution, I provide a content account for “what” is transferred from the domain of biological solutions to the domain of problem formulation. Following from this description, I propose a process theory of problem evolution in design, called PE.BID, which can be broken into components: problem formulation; design problem goals; design problem strategies; problem transformations, and an integrated problem-solution memory. I provide theoretical and qualitative support for each component.

1.2.5 Support Tools for BID

In this section, I will review the four-box method of problem formulation, a tool that addresses the challenges of problem definition in the BID classroom. In the *in situ* studies conducted in 2006 and 2007 I found that students experienced considerable difficulty in formulating design problems of their own creation. Since design problems provide both index and evaluation criteria for the biological analogies, poorly defined problems yield additional challenges, including difficulty searching for and evaluating analogies. While process and tool support were provided to assist student designers with the task of search, problem formulation and analogy evaluation remained unaddressed. To address this challenge I implemented the four-box method of problem formulation, which is based on the SR.BID content account. The four-box method of problem formulation was extended to analogical evaluation through a tool called the T-chart method of analogical evaluation. Based on the success of these tools, SR.BID was tested as an underlying framework for distributed knowledge acquisition for biologically inspired design, through

a web-based application.

The implementation of the four-box method for problem formulation is assessed in three ways. First, after students are trained on the method, they are provided with an assignment which requires the use of the method. The ability of students to use the tool after a single training session is measured in terms of the *accuracy* with which students are able to use the method to define. The students continue to use the four-box method throughout the class, extending its use to include both problem formulation and analogical evaluation. After using the method for several additional weeks, students are asked to reflect over their use of tools and methods used in the classroom. The results of this study, including both the four-box method of problem formulation and T-chart method of analogical evaluation are reported in terms of a qualitative assessment of these student reflections. Finally, the four-box method is implemented in a web-based application. The web-based application demonstrates in principle how students can apply the four-box method to generate structured knowledge about design problems and biological systems with minimal cost.

1.2.6 Research problem 3

Problem formulation in BID plays an important role in searching for and evaluating analogical sources. However, we observe that students struggle with problem formulation in BID, and consequently with analogical evaluation. No tools exist to support problem formulation or analogical evaluation in BID.

1.2.6.1 Research Question 1.1

To what extent can SR.BID be used accurately for the design task of problem formulation in the context of the BID classroom?

1.2.6.2 *Hypothesis I.1*

The four-box method (shown in Figure 1-4) can be used accurately by all students to represent design problems in BID.

Operational Environment	Functions
Specifications	Performance Criteria

Figure 1-4 Student designers use the four-box method of problem formulation to articulate a problem in terms of (1) operational environment, (b) functions, (c) specifications/constraints and (d) performance criteria.

1.2.6.3 *Method I.1*

The four-box method is implemented in 2011 and 2012 within the existing framework of the class as a replacement for generic problem definition assignments at the individual and team levels. After training in the four-box method assignments are collected and evaluated in terms of number of student assignments completed, and accuracy with which the method is used. Students are provided with a survey at the end of the semester which seeks to understand opportunities for improvement in the four-box method. They are provided with a take-home final reflection assignment, which prompts for open comments about the 4-box method. Conclusions are drawn from student use data, surveys and reflections with respect to the feasibility of the system for systematically encoding problems in biologically inspired design, as well as for design improvements for future implementations.

1.2.6.4 *Study Results I.1*

Completion rate on the 4-box portion of the assignment was greater than 95%, providing evidence in favor of students' ability to complete the task. The overall accuracy of student use of the four-box method, excluding the three students that did not complete

the assignment, was measured over a total 1058 concepts. For all concepts across both years and all concept types, average student accuracy was 85.6%. There was no statistically significant variance in accuracy across years, gender, major, or method of reporting. There was a variance in accuracy across concept types; where operational environment concepts were more accurately used, while constraints and specifications were used less accurately.

1.2.6.5 *Research Question I.2*

How are the SR.BID and four-box representations used in the context of biologically inspired design with respect to the task of problem definition?

1.2.6.6 *Hypothesis I.2*

The T-chart method of analogical evaluation can be used accurately by all students to evaluate and compare analogies, and to provide support for explaining why analogies were selected.

1.2.6.7 *Method I.2*

SR.BID and four-box representations were used in a number of assignments in class from week 7 until the end of the class. SR.BID and the four-box method are also included in the T-chart method for analogical evaluation (Figure 1-5). The T-chart method of analogical evaluation generates a four-box model for the design problem (left column) and for the biological system (right column) which can then be compared side-by-side. Students are encouraged to consider the implications of differences and similarities in their evaluation of the analogy.

Design Problem		Biological System
Operational Environment		Operational Environment
	Different	
	Similar	
	Same	
Functions		Functions
	Different	
	Similar	
	Same	
Specifications		Specifications
	Different	
	Similar	
	Same	
Performance Criteria		Performance Criteria
	Different	
	Similar	
	Same	

Figure 1-5: In the T-chart method of analogical evaluation, designers compare a design problem to a biological system across the four-dimensions of the four-box method, using a simple comparison: different, similar, or same.

As a final exam, students submitted open reflections on their experiences with the tools and methods they were taught in class. The reflections were guided such that a student need not discuss the four-box method or SR.BID representations, although most of them did. Of the 34 students, all students reflected at least on either SR.BID, the four-box method or the T-chart method. Reflections were summarized and coded. Table 1-6 shows of positive and negative comments, summarized by category, associated with the four-box method. From this data I can infer that students were more positive about the method than negative by a wide margin (nearly 3 to 1), and that they found it valuable for many of the reasons I anticipated – problem definition, clarification, and breakdown, focus and organization. Student comments reflect similar value association with the T-chart representations for the purpose of evaluating analogies.

Table 1-6. Positive and negative comments associated with the 4-box method of problem formulation. Comments are gathered from student reflections at the end of the 2012 BID class, and are summarized by category.

Positive Comments	29	Negative Comments	10
Define/specify/clarify problem	8	Decrease/Limit creativity	3
Breakdown, problem	6	Limited to a single environment	2
Focus	2	Confusing, categorizing concepts	1
Organize data/knowledge/problem	2	Confusing, redundant	1
Search, aid	2	Confusing, specification v performance criteria	1
Understand, system	2	Difficult to learn, to use initially, different at first	1
Analogy, matching	1	Increased workload	1
Direct inquiry	1		
Easier than another system	1		
Evaluate, problem	1		
Understanding, SR.BID	1		
Useful, operational environment	1		
Visualization	1		

The first four of the comments on the positive side provide *strong evidence for the value of the four-box method of problem specification*: in particular that, as intended, *it provides students with a greater capability to define/specify/clarify, breakdown, organize, and otherwise grapple with complex design problems.*

Encouraged by the successful implementation of the four-box and T-chart methods in class, I look next to addressing the problem of scalable knowledge engineering in the context of BID.

1.2.7 Research problem 4

From observations of systems implemented to support BID, we know there is a tradeoff between representational complexity, cost of representation and potential value/tasks supported. Existing support systems do not yet provide a sufficient

return on investment such that system builders are able to get widespread adoption. How might these systems acquire structured data on thousands or tens of thousands of biological source analogues that would be necessary to drive value to the design users?

1.2.7.1 Research question I.3

To what extent can SR.BID be used to support design processes in BID over a distributed (e.g. web-based) platform?

1.2.7.2 Hypothesis I.3

The SR.BID for Web Application can be used to support designers for problem definition, biological analogy building, and analogical search and evaluation, over a distributed team-based platform in the context of BID, with minimal additional investment over current assignment workload.

1.2.7.3 Method I.3

The SR.BID Web Application is developed and deployed over a short window in the context of the BID classroom. Students are provided with a brief 20 minute training session in class, and encouraged (by the researcher) to voluntarily use the system to enter problem definition information, and biological source information. Student interactions with the system are recorded in a database to determine the amount and type of interaction students engage in with the tool. Timing of transaction data is analyzed to determine the length of time for completion of desired tasks, such as entering a new biological system or a problem definition. Table 1-7 provides a sample of a transaction report for a single user.

Table 1-7 Sample of transaction reports for a single user for the SR.BID Web Application.

USER ID	DATE & TIME	TRANSACTION TYPE	TRANSACTION
6	11/8/2012 19:52	NEW PROJECT	Added Project: The Signal Seed
6	11/8/2012 19:56	NEW REFERENCE	Added Reference: Project document, final report.pdf
6	11/8/2012 19:57	NEW REFERENCE	Added Reference: Project document, materials assessment.pdf
6	11/11/2012 0:05	NEW REFERENCE	Added Referenc: Project image, images of prototypes 1 & 2
6	11/11/2012 0:07	NEW BIOLOGICAL SYSTEM	Maple seed (samara)
6	11/11/2012 0:07	NEW PROJECT	Maple seed (samara)
6	11/11/2012 0:13	NEW REFERENCE	Added reference: samara seed distributio.pdf
6	11/19/2012 11:29	NEW REFERENCE	Blank entry
6	11/19/2012 11:30	NEW REFERENCE	Added Reference: Project document, QA.pdf
6	11/19/2012 11:30	DELETE EXTERNAL FILE	Deleted blank entry

By analyzing the transaction reports, I determined that in practice, student designers can use SR.BID and the four-box method of representation to build complete structured knowledge representations of both design problems and biological systems in the context of the class on biologically inspired design. Moreover, such representations can be entered into the system in comparatively little time; between 20 and 40 minutes for a complete model. This, combined with the students' reported value from the use of these methods in class, suggests that the system can be used to build a distributed, joint problem-solution database for the support of BID.

Moreover, the realization of joint problem-solution models in a database organized according to the SR.BID framework provides proof-in-concept that SR.BID can be used to organize and instantiate a memory capable of performing analogical problem

evolution. This provides a computationally plausible support argument for the process model of memory developed earlier.

1.2.7.4 *Summary of Section*

We know from the previous studies in this section (studies I.1 and I.2) that student designers are capable of learning the four-box method of design problem description quickly, applying it to new design problems with greater than 80% accuracy in most cases, and that they find value in the methodology for ordering and organizing their thinking about their design problems. Moreover there is some evidence in support of its use in analogical evaluation.

We also know from this study that individual student designers can use a web-based platform enter this information into a distributed database of problems and biological systems, and that in this prototype system they can generate meaningful, multi-model descriptions of design problems with an investment of less than 60 minutes. From this data, *I claim that viewed as an underlying scaffold for both tools and technology, SR.BID and the SR.BID Web Application can be used for low cost, massive distributed collection of design problem and biological system information. This information provides value to designers for tasks specific to biologically inspired design, including design problem formulation, and analogical evaluation.*

1.3 Personal Motivation

In this section I briefly review my personal motivation for exploring the problem of analogical problem evolution.

1.3.1 My observations of BID class

I have always been fascinated with biology, and prized the opportunity to learn about

biology through the eyes of design. After witnessing in the context of the BID class at Georgia Tech the use of biology to generate new technical solutions for interesting projects, I believed that this design paradigm could truly change the world. However, I noticed that instructors and students struggled with critical key aspects of the class.

First, students struggled with the formulation of design problems. This took me by surprise. Georgia Tech is one of the top engineering universities in the country! How could engineering students struggle...and fail...at a task as fundamental as defining a design problem. The mystery was afoot! Additionally, I saw in project after project, biological solutions which students considered interesting or cool always had a way of fitting “just right” into their design problem. There was a game being played here...somehow, problems and solutions were moving toward the same goal.

Second, while students often quickly produce biologically analogies to match a design problem, and could on the fly generate new solutions to problems using these analogies, when asked *why* these analogies were good, students started back with blank faces. Students had an intuitive notion of why an analogy matched a problem, but when pressed they had no framework for articulating why. Something else was going on here.

Both problem formulation and analogical explanation are real problems faced by designers, and central to the issues of biologically inspired design. I thought, understanding these issues may provide me with an opportunity to help a new, developing community that I believe can have a significant impact on the way we design.

1.3.2 Parallels with PhD research

The more I progressed on my PhD journey, the more I recognized that the struggle that the undergrad students had in defining a good design problem was not as uncommon

as I thought. Identifying a design problem in a world filled with potential problems is difficult. First what problem to pick? In innovative design, the problem has to be interesting. It must be a problem people care about, something not yet solved, and something challenging, but with enough research behind it to make solving it tractable. Second, it is not enough to identify the problem; one must understand it deeply in order to solve it. Student designers find they get a handle on the problem quickly, they race to a solution, and then something unexpected happens, it doesn't work the way they want it to, or their customer (the Professor) thinks there might be a better way. There is always something unanticipated, something more to understand. So they learn from your mistakes, you research the problem, you grasp at the nuances, you study what has gone before and what worked and didn't. Now you can try to solve it...and again it blows up. Solutions are always blowing up. Persistence through failure, I think, must be the number one trait of any inventor.

I'm not just talking about student designers anymore, but about myself and this research as well. PhD problems have many parallels to design problems. Thus as I was studying the phenomenon, I was living it as well. All the more reason to find a solution!

1.4 Limitations and Assumptions

The limitations of my research are that: (1) All data for this research is collected within an undergraduate class on biologically inspired design, which changes, often dramatically, in format, style and content from year to year. I assume the phenomenon of analogical problem evolution as observed and described is based on an underlying process that is consistent across these variations. (2) While the class represents a new domain of formal study, I assume that the design actions of biologically inspired design

in this class are more broadly representative of design actions of biologically inspired design. (3) The designs studied and the resulting phenomena described occur distributed in time, over the course of weeks or months, and space, wherever students are able as well as electronic communication, I assume that instruments used to collect the data, mainly homework assignments, represents the honest work of students, and is not intentionally misrepresentative. (4) I assume the designers in the class are all student designers, and are representative of the BID student design population consisting of students mostly in mechanical engineering, biology, biomechanical engineering, and systems engineering; I assume that findings will generalize at least to the population of student designers with majors largely represented by this class; (5) I assume that the qualitative method of ontologically grounded theory, and resulting differential analysis to determine changes in problem formulation, are sound; I likewise assume SBF models of analogy and CPS characterizations of problems are sound; (6) Generalizations from this study are circumscribed by the class of generalizations that may be drawn from the case study methodology.

1.5 Delimitations

The delimitations of the research are: (1) that only student designers are selected for study; this expert designers, engineers or biologists are excluded from this study; (2) only the classroom context is studied; design experience outside of the context of a classroom is excluded from this study; (3) the raw data concerning problem formulations are representations created for and in the context of homework assignments; data concerning problem formulations outside of the context of homework assignments is not used; (4) that construction of a model using ontologically grounded theory, with reliability

statistics >80% are valid; models and coding constructed with less reliability must be discarded or changed and retested over new data.

1.6 Organization

This dissertation is organized around the three core themes: design problem formulation, design problem evolution and support for design in BID. Prior to discussing the core themes, in *Chapter 2*, I will situate the work in a review of related research. In *Chapter 3*, I will establish the context in which these studies occur, both in terms of the domain of design, and the specific classroom context. I will also describe the research design for the overall dissertation which consists of a series of studies conducted under a larger case study framework. Subsequent sections will describe the methodologies used for specific individual studies contained in those sections. In *Chapter 4*, I will establish the phenomena of interest by describing the 2006-2007 exploratory studies and the key findings relevant to this dissertation. Chapter 4 will end by raising the first two research problems I address in this dissertation. In *Chapter 5*, I will address the first core theme: design problem formulation. First I will assess the relative strengths and weaknesses of existing theories with respect to design problem formulation in the context of BID. Using SBF as an ontological seed, I will use ontologically grounded theory to develop a content model of design problem formulation in BID. In *Chapter 6*, I will address the second core theme: design problem evolution. After providing a richer description of analogical problem evolution based on SR.BID, I will postulate a more general process theory of problem evolution, PE.BID. I will provide evidence in support of each of the fundamental concepts in the PE.BID model. In *Chapter 7* I will discuss the implementation of three tools for design support, based on my theory of SR.BID. In

Chapter 8, I will situate this research in the broader context of cognitive science, design science, human-centered computing, and biologically inspired design. At the end of this section, I will conclude this dissertation with thoughts on future research.

2 RELATED RESEARCH

In this dissertation I provide an account of design problem-solution representations and an account of the process of analogical problem evolution. These concepts are reified in a domain, the nascent but rapidly evolving field of biologically inspired design. The background research for this undertaking therefore spans problem formulation in design, content and process accounts of design problems and analogies, process accounts specific to biologically inspired design and specific to problem-solution coevolution, and finally we round the research out with a review of existing applications for the support of biologically inspired design. While this dissertation is not focused on the deployment of technology per se, the last topic is important for understanding the implications of the research.

2.1 Problem formulation in design

2.1.1 Defining design as an ill-structured problem

Simon (1973) categorized design as an ill-structured problem. That is, design as a class of problems are inherently under specified and include uncertainty not only with respect to the proper end result, but also with respect to what method(s) might be applied to achieve a result. Even the result itself is subject to uncertainty, in that one may never know whether a design is optimal in an absolute sense. This was similar to Rittel's (Rittel & Webber, 1973) concept of wicked problems. The categorization of design as an ill-structured problem extends as far back as Reitman (1964) who highlights the under-specification of design problems. Dorst (2003) discusses three degrees to which a problem may be ill-defined: (1) some aspects are *determined* by hard requirements, (2)

the major part is *underdetermined*, subject to the design choices made by the designer during the process, and (3) some aspects are completely *undetermined* and subject to the style of the designer. In modern design, that design problems are ill-defined is accepted as given; as Cross (2001) put it in his review of 30 years of design studies, “It is widely accepted that design ‘problems’ can only be regarded as a version of ill-defined problems.”

2.1.2 Two schools of thought on problem structure

There are two core schools of thought on how the structuring of ill-defined design problems is approached. The first school follows the initial work on rational problem solving from Simon (Simon 1973, Newell and Simon 1972). In this view, a problem constitutes a search space, which can be broken down into independent sub-problems, where the sub-problem can be systematical searched until a sufficient solution is identified. The component solutions to sub-problems can then be synthesized into an overall solution. Thus, a problem is first structured, and then a solution is synthesized. Goel and Pirolli (1994) show through a protocol study of design this two-phased approach between problem structuring and solution development, and distinguish it as fundamentally different from other problem-solving activities, which neglect problem structuring almost entirely. Gero (1993) extends the problem-space and search metaphor to suggest that exploration in design is a process which creates new design state spaces or modifies existing state spaces, extending the amount of space which can be explored for design. As Dorst (2003) points out, this rational problem solving approach is representative of the positivist epistemology, suggesting that a problem exists independent of the problem-solver, and can be analyzed and studied objectively, yielding

to systematic, scientific processes.

The other school of thought stems from Schön (1983) and what he calls reflective practice. From this viewpoint designers subjectively frame a problem often in conjunction with the generation of one or more possible solutions. While Schön does not explain how such framing occurs, he ties together the inherent subjectivity of the problem as viewed by the designer with the notion that solutions provide a fundamental lever in framing the problem. As he states in later work “problem solving triggers problem setting (Schön 1988).” Dorst (2003) equates this perspective to the phenomenological paradigm in which the construction of reality, in this case the design problem, is inherently subjective. As Dorst and Cross (2001) observe “...designers did not treat design assignments as an objective entity. All designers interpreted the assignment quite differently in awareness of their own design environment, resources and capabilities.”

Since the development of these two schools of thought, many case studies, protocol analyses, and performance tests have been conducted usually through one lens or another. Cross (2001) provides a comprehensive summary of 36 studies conducted in the thirty years spanning 1970 – 2000. Relative to problem definition, certain key points of debate have arisen, which can be traced back to the difference in schools of thought on design problems.

2.1.3 Top-down design

Standard prescriptive methodologies for mechanical engineering and software design (Wirth 1971, Dahl, Dykstra, and Hoare 1972, Roozenberg & Eekels 1995, Pahl & Beitz 2003) suggest a top-down analysis of the problem using function decomposition strategies, for instance. In such a case, in each step of the process detailed design

decisions are deferred until the entire problem is sufficiently refined in abstraction. Such processes, while acknowledging the iterative nature of design, suggest that an initial problem formulation can be divorced from solution and analyzed objectively prior to solution instantiation. These methodologies are firmly rooted in the rational problem solving school of thought. Such top-down problem definition strategies are considered as rational, disciplined and well-behaved design (Guindon 1990).

Guindon (1990) observed the practice of a small number of software designers, and showed that rather than applying perfectly top-down strategies, software engineers are more “opportunistic”, and will occasionally be seen to solve a particular sub-problem, prior to returning to the top-down, breadth first problem structuring activity. Likewise, Chadrusekaran (1990) notes that while hierarchical functional decomposition of a problem is an important design task, “in many domains, constraint generation of some sub-problems alternate with partial designs of others, which in turn provide partial constraints for yet other sub-problems.” Ball, et al (1999) counter that, rather than “opportunistic” or “ill-behaved” designers are simply performing a top-down, selective depth-first search which is being invoked to validate the high-level design concept where a designer is unsure. Novices, as one might predict, tend to perform more depth-first problem solving than experts who, being more certain tend to provide breadth-first problem structuring. In 2004, Cross counters with models from Holyoak (1991), Adelson & Solway (1988) and Cross & Clayborn (1998) in which he claims experts do not conform to breadth-first, top-down strategies.

2.1.4 Solution-orientation

Contrasted with traditional top-down design processes are design studies that show

designers often begin with solution conjectures first. Lawson (1979) formalized design as either problem-driven or solution-driven, and characterized the later as being more characteristic of design-based problem solving. The so-called solution-oriented approaches stand in contrast to the top-down, problem-oriented approaches. In such solution-oriented approaches designers quickly conjecture partial solutions to problems, with very little problem structuring or definition occurring prior. Analysis of the proposed solutions can then be used to contextualize and more deeply understand the problem. Hillier, Musgrove, and O'Sullivan (1972) describe the theory of *conjecture-analysis*, which matches observations in architecture and in which early solution conjectures are seen to rapidly reduce the search space by eliminating incongruent alternatives. Darke (1979) expands this theory to *generator-conjecture-analysis* in which a “primary generator”, which can be an idea, or set of coupled ideas, is used to both narrow down the search space and to provide a starting point for the designer. The “primary generator” is imposed on the design problem by the designer. In terms of reflective practice, this primary generator can then be used to frame the problematic design; “set its boundaries, select particular things and relations for attention, and impose on a situation a coherence that guides subsequent moves (Schön, 1988).”

There are however, some potential drawbacks to the solution-oriented approach. Restrepo & Christiaans (2004) show that commitment early, as with solution-oriented approaches, can lead to design fixation. This is not unexpected, as Rowe (1987) observed that “a dominant influence is exerted by initial design ideas on subsequent problem-solving directions...a considerable effort is made to make the initial idea work rather than adopt a fresh point of departure.” This may also be a function of when requirements are

produced. Restrepo & Christiaans (2004) show that problem oriented designers produce their requirements throughout the entire session, whereas solution-oriented designers specified their solution at the beginning of the process. Kruger and Cross (2006) show that for the same problem, some designers employ such solution-oriented approaches, while others use problem-oriented approaches. For their experiment they show that solution-oriented design still tends to produce creative results, but lower overall quality.

Note this notion of solution-orientation is linked to a **conjectured solution**. In this dissertation I delineate between two different classes of solutions, and thus two different kinds of solution-based design. The first class, **the conjectured solution**, is the solution class with which most existing theories of design concern themselves. In all instances of design theory mentioned, solution-oriented processes are so-called because they include an early conjectured solution. The majority of theories that cite early solution conjectures discuss relationships between the evolution of the problem and this conjectured solution. The second class of solution, and the class with which I am concerned, is the class of **solutions that currently exist**, whether as a solution to the at-hand design problem or a solution that appears unrelated, such as a biological system. My concern is with how such **existing solutions** interact with the design problem. In particular, how **existing solutions** outside of the traditional domain of the design problem – existing biological solutions relative to engineering problems – influence the formulation and evolution of the design problem.

Maher, Poon and Boulanger (1993) used a slightly different term for the relationship between problem and solution. Rather than classifying the processes as problem- or solution-oriented, they describe the process as a co-evolution. They used the concept of

genetic algorithms for a well-defined design problem, to show how such co-evolution could occur computationally. The use of a genetic algorithm required a routine and structured problem definition, limiting the degree to which the technique could generalize, but it served as an interesting proof of concept. Figure 2-1, adapted from their work shows how starting a problem $P(t)$, a conjectured solution $S(t)$ is generated. This conjectured solution generates new information (shown as a small blue box), which is then transferred to the designers' understanding of the problem, generating problem $P(t+1)$. This new problem definition, in turn, is used to generate the next solution iteration $S(t+1)$. The process can iterate until the solution sufficiently meets the requirements in the current problem state.

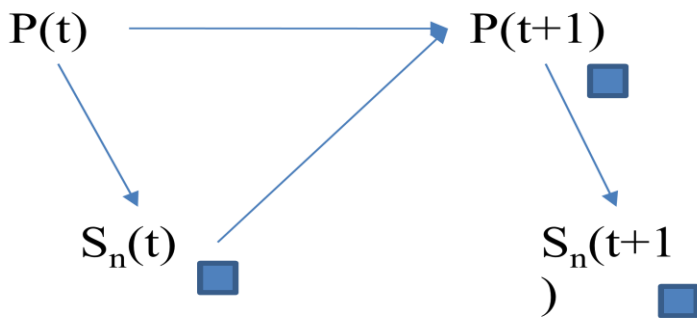


Figure 2-1. Problem-solution evolution, Maher, Poon and Boulanger (1993)

Subsequently, Dorst & Cross (2001) examine and elaborate on the notion of co-evolution and include in their definition the concept of partial structuring of the solution and problem spaces. In this way, sub-problems could be defined and solved, and the information thus gathered could cycle back through the problem description as partial evolutions of problem and solution. Figure 2-2 shows the process developed by Dorst &

Cross. In this case, the triangle in $P(t+1)$ represents the development of a sub-problem structure, from which a sub-solution $S(t+1)$ is developed. The information from the development of that sub-solution is then cycled back to inform further developments in the problem. Note once again, these are conjectured solutions that are being evaluated in the context of the problem and informing the problem development. The process of sub-problem creation, solution generation, and new problem formulation occurs until all sub-problems are solved sufficiently by the existing array of arranged sub-problems. As described by Dorst and Cross, the process continues until a bridge is built between solution and problem such that the solution to the existing problem is apparent. The metaphor implies the bridge is extended from each side (problem and solution) of the gap, until it makes a solid connection somewhere in the middle.

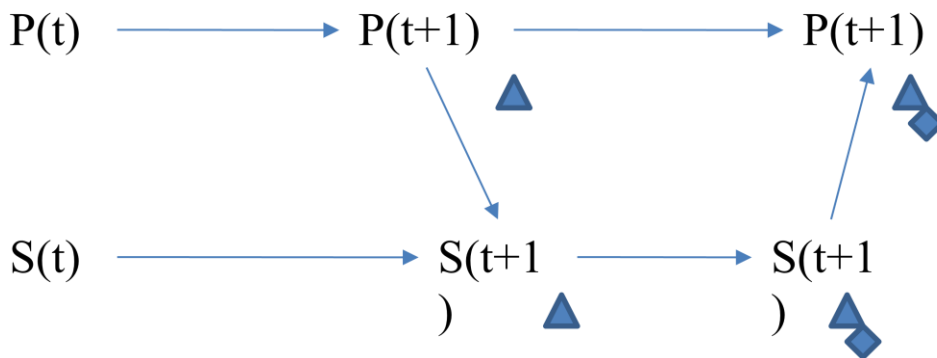


Figure 2-2. Problem-solution evolution, Dorst & Cross (2001)

While this process reflects the design process, there is a parallel with the work of Nersessian (2009), in which a scientific problem is understood in terms of a simulation. A simulation represents the embodiment of a conjectured solution about the scientific problem at hand. The simulation evolves interactively with the understanding of the

scientific problem similar to what is seen in design. While the analogy does not hold perfectly – design is intended to change the world, scientific inquiry to understand it -- there is a striking similarity in the iterative processes seen here.

In analogical problem evolution, existing solutions also influence problem development. As shown in figure 2-3, the modification I make to the process of problem-solution co-evolution is the inclusion of an existing solution that exists at time (0), $S_e(0)$, where the e stands for existing. In subsequent stages the subscript n stands for a conjectured (new) solution. As before, the solution may iterate until solution and problem match occurs. In this case, however, a conjectured solution is not necessary at every step. At any time an existing solution can be introduced into the process that lends insight into the current problem, transforming it, and opening up potential new solution paths. This provides a high-level model of what I call analogical problem evolution.

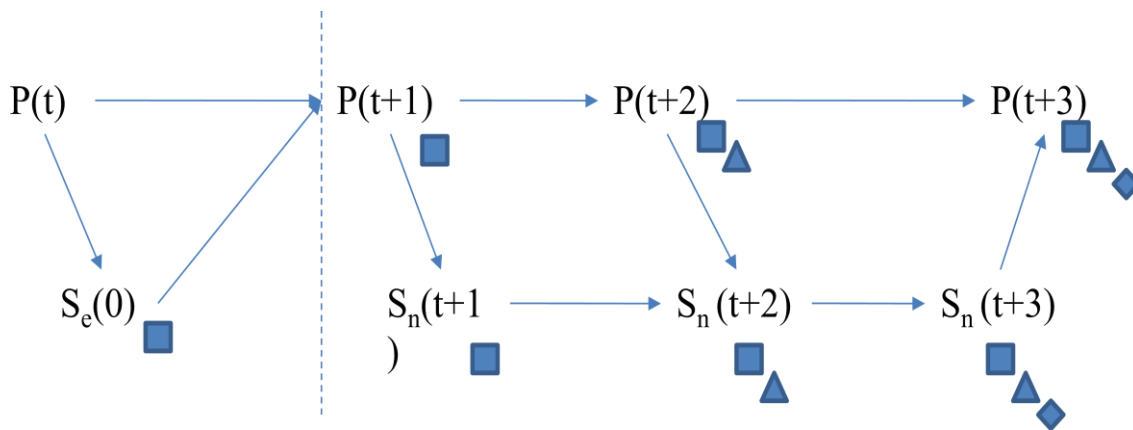


Figure 2-3. Analogical problem evolution

2.1.5 Problem Decomposition in Design

Problem structuring through decomposition into sub-problems occurs in both solution- and problem-oriented approaches. In prescriptive methods such as Pahl & Beitz (2003), the functional decomposition of a problem into functional sub-problems is an explicitly defined part of the design process. However, some studies show that such decomposition seems to happen even without conscious direction.

Ho (2001) documented the use of both implicit and explicit decomposition of problems. That is, in the observed verbal protocols designers often provided problem decompositions – in this case, a mixed combination of functional and component (form) decompositions – without verbally indicating an intention of creating the decomposition. They note that such implicit decompositions resulted from working forward (depth-first) strategies that were engaged on solving a sub-problem. Then working backward, feedback from the results of this working forward strategy directed the next sub-problem to be considered, again, implicitly.

Liikkan & Pertulla (2009), based on the work of Ho in 2001, applied the implicit and explicit decomposition strategies to their analysis of a group of mechanical engineering students in a controlled experiment setting. They note that approximately 1/3 of the utterances made by designers were problem-oriented, and approximately half of those could be traced to implicit decomposition. They put forth a high-level cognitive model based on their observations that suggests that implicit decomposition occurs during problem interpretation activity, which explicit decomposition occurs during solution generation activity. They note that explicit decomposition, which occurred only twice, had no correlation with the quality of results.

Additionally, they claim that such implicit decomposition is driven from a library of pre-existing decompositions possessed by the designer. The amount and ability to match such relevant decomposition knowledge with an initial problem is dependent on the size of the internal library of decompositions the designer has access to; since novices have smaller internal libraries than experts, they more often producing incomplete or unfit decompositions. This finding coincides with the earlier work of Lloyd & Scott (1994) who posit that domain experience leads to the ability not simply to develop a design, but to structure and decompose a design problem. Restrepo & Christiaans (2004) suggest that while the creation of design requirements is triggered by prior knowledge, it is also triggered by knowledge acquired during design by interaction with the solution or with external sources of information.

2.1.6 Time Spent on Problem Structuring

Restrepo & Christiaans (2004) show that information gathered for the purpose of problem structuring, for example about users or the environment, requires additional interpretation and manipulation before it can be used by the designer. This is in contrast to gathering solution information, for example material specifications, the application of which is a known process for the designer.

Christiaans (1992) suggest that “the more time a subject spent in defining and understanding the problem...the better able he/she was to achieve a similar result.” However, from a controlled experiment conducted of 53 engineering students, Atman & Chimka (1999) show that for freshman students design quality is inversely proportional to the amount of time spent on problem definition, whereas it is positively correlated with the amount of time spent in evaluation and decision making. According to Atman &

Chimka apparently “some of the freshman students seemed to ‘get stuck’ defining the problem.” On the other hand, for seniors, the amount of time spent in problem scoping highly correlates with the number of constraints their final design satisfied. Seniors also asked for more information during the design process. This suggests that experience plays a significant role productive problem structuring. As Cross (2004) states, “it appears that successful design behavior is based not on extensive problem analysis, but on adequate problem scoping and on a focused or directed approach to gathering problem information and prioritizing criteria.”

2.2 Content accounts of problem representations

While major design theories assume the existence of both design problem and design solution representations, they tend to posit rich content accounts of solutions, content accounts for problems are relatively impoverished. Prescriptive accounts from the engineering field offer richer tools for gathering requirements, but provide little in the way of cognitive content accounts per se.

2.2.1 Functional representations of design problems

Functional representations (FR) were developed to provide a top down representation of a device, as a set of function/sub-function relationships. While these representations focus on existing or designed artifacts and not on problems per se, these representations touch on problem formulation and/or imply problem representation strategies. Functional modeling, a prescriptive technique usually considered part of conceptual design offers another method for defining a design problem. The systematic approach of Pahl and Bietz (Pahl and Beitz, 2003) advocates function-flow diagrams as a means of decomposing a design problem. In such a diagram, a design is conceptualized as a set of interconnected

functions. Each function transforms a specified set of inputs into a specified set of outputs, which in turn may be passed to another function as inputs. In this case, the design problem considers externalities to the system as the set of initial input flows into the system. Several taxonomies of flow and function types exist, for example Altshuller (1984) provides a list of 30 functional descriptions, Sturges et al. (1996), Kirschman and Fadel, 1998 Hundal's Function database (1990), the Function Basis of Stone and Wood (2000). While these systems of categorization have traditionally been applied to the systematic breakdown of small, existing solutions, e.g. a hair dryer, in principle they could be applied to functional descriptions of problem specifications.

The Situated Function-Behavior-Structure (Gero, 1990, Gero and Kannengiesser, 2004) account of design offers a description of problem as an initial set of requirements R taken from the external world, from which the process of Formulation produces interpreted functional (F_i), behavioral (B_i) and structure (S_i) variables and constraints. While neither the characteristics of the requirements nor the process of Formulation are well defined, this high level view of problem representation attends to the interpreted nature of functional requirements; that is for each designer (or interpreter), a set of requirements may be translated differently.

The Structure-Behavior-Function (Goel, Rugaber, & Vattam, 2009) account of design likewise does not itself offer a description of a design problem per se, but an implicit definition is embedded in the teleological nature of the definition of function: that is, a function is the desired or intended goal of the designer. This interpretation of function is widely used for instance by Umeda et al. (1996), Umeda and Tomiyama (1995), and Chandrasekaran and Josephson (2000). The AI systems built using SBF (Goel &

Chandrasekaran 1989, 1992; Goel, 1992; Goel et al., 1997, Bhatta & Goel, 1994, 1997; Goel & Bhatta, 2004) confirm this interpretation by providing to-be-solved problems in terms of a functional specification. The specification of function in SBF includes a given state, a resultant state and external stimulus. External stimulus may serve to incorporate salient factors from the operational environment. The problem can be constrained further through limitations in the components (structural elements) to which the designer or AI system has access.

Chandrasekaran and Josephson (2000) describe a problem as a “functional specification for the design task”, where a specification is given as an environment and a set of behavioral constraints for the environment. It is the task of the designer to create a device that in some mode, meets the established constraints in the environment. They distinguish their definition from the traditional definition of design which is to generate a device to satisfy a set of behavioral constraints, by adding that the device must also do so in a manner embedded in a particular environment.

2.2.2 Design problem formulation in practice.

Likewise, Pahl & Beitz (2003) in the field of mechanical design propose the establishment of a list of requirements during problem framing. They suggest classification of requirements based on “demands” versus “wishes” and a tabulation of both qualitative and quantitative criteria. They provide the following checklist of 16 requirements types: geometry, kinematics, forces, energy, material, signals, safety, ergonomics, production, quality control, assembly, transport, operation, maintenance, recycling, costs, and schedules. Hundal (1990) likewise provides a general specification which “should be abstracted to solution-neutral terms.” The specifications are broken into

(a) proposed and existing systems, which considers constraints from both the existing design and the context of systems into which the designed system is to be integrated; (b) importance of the requirement as either a “demand” or a “wish”, (c) lifecycle specifications including planning and design, production, marketing, product use, and scrapping/recycling; and (d) type of requirements including engineering/technical, economic, ergonomic, legal and other. Functional requirements (what the system must do) and operational requirements (how the system must do it) “play the most vital role” in the conceptual design phase.

2.2.3 Analytic representations of problems

Recently Dinar, et al (2011, 2012) looked at representations that can be used to compare problem formulations in design. They establish a Problem Map framework with five general categories of problem concepts, based principally on the Gero’s Function-Behavior-Structure model. The five categories are Function, Behavior, Artifact (Structure), Requirement and Issue. A problem map is the set of states of these concepts and their relationships to each other at a point in time. One can compare problem maps from one state to another to determine changes to problem maps over time. They further amplify their work by delineating a set of transformation types that may be carried out on a problem map to show differences between one state and the next, for instance a “decomposition” transformation that adds child nodes to a parent node. While the goals, context and details differ, their thinking is similar in principle to this dissertation work and serves as additional validation for the need for more robust models to analyze problem change over time.

2.3 Design process accounts specific to biologically inspired design

The theories discussed thus far are general theories of design. In this section, I discussed theories of design developed specifically to account for certain observations of the practices of biologically inspired designers.

2.3.1 Problem-driven and Solution-based design.

During in situ cognitive studies conducted in 2006 and 2007, we observed the existence of two high-level processes for biologically inspired design based on two different starting points – *problem-driven* and *solution-based* (Helms, et al 2008). Here we use the term solution-based to describe a process that begins without a particular problem in mind, and where starting point for ideation is a biological source system. The term solution-based or solution-driven design has been used alternatively to describe design processes that propose solutions prior to a deep analytical phase (Krugar & Cross, 2006), and in biologically inspired design to describe reverse-engineering and application of a biological solution (Wilson, 2008) to a given problem. Similar account of solution-based design can be inferred from prescriptive accounts of biologically inspired design, such as Biomimicry Institute’s “Biology-to-design” design spiral process (Biomimicry 3.8 Institute, 2013). This process of problem-driven design is an instantiation of the cognitive process of analogical reasoning (Clement 2008; Dunbar 2001; Gentner 1983; Gick & Holyoak 1983; Goel 1997; Hofstadter 1996; Holyoak & Thagard 1995; Keane 1988; Kolodner 1993; Nersessian 2008). Solution-based design appears new and different from the perspective of design theory, which is traditionally problem-driven (e.g., Dym & Brown 2012; French 1996; Pahl & Beitz 2003). Thus, the BID course acts as a research laboratory for developing identifying and studying new BID constructs and processes.

2.3.2 Solution-based Biologically Inspired Design Process

Whereas the normative biologically inspired design process taught in the class was problem-driven, we observed that in practice the design process often began with a biological solution. Some classroom exercises, and many case-studies of biological design, began with a biological solution, extracted a deep principle, and then found problems to which the principle could be applied. In general, the solution-driven biologically inspired design process follows the steps listed below.

Step 1: Biological Solution Identification

Step 2: Define the Biological Solution

Step 3: Principle Extraction

Step 4: Reframe the Solution

Step 5: Problem Search

Step 6: Problem Definition

Step 7: Principle Application

The process of solution-based design provides many clues about how problems and solutions might be organized in memory, and how they must interact with each other. As a result of the solution-based design process we know that a solution must have some “hooks” into problems; not just the problems the solution solves, but also the ability to access and modify other problems. Since this process is so heavily influenced by solutions, and since it represents so many of the observed cases of biologically inspired design, it seems natural to attempt to extend solution-based problem evolution to account for this process as well.

2.3.3 Compound Analogy

Solving complex problems by decomposition where designers break complex problems into less complex ones is not new. But when we make the decompositions explicit in the context of analogical design, it becomes apparent that the processes of decomposition and analogy influence each other. We describe their interplay as *compound analogical design* (Helms, Vattam & Goel, 2008).

In the simplest case of compound analogical design, when a target design problem is presented, the designer iteratively decomposes the problem into sub-problems to create a problem abstraction hierarchy. The problem may be decomposed along functional lines, although we have observed other lines of decomposition (temporal, structural, etc.), often intermingled. Assuming that the problem is decomposed along functional lines, each node in this hierarchy of decomposition is a function to be achieved. Each function (node) can be used as a cue to retrieve known solutions that achieve that function. Solutions are transferred to the current problem, and aggregated to generate the overall solution. This process explains complications that often arise during reintegration, as the solutions from disconnected analogies may not integrate cleanly at their boundaries, or may have overall constraint mismatches.

In many cases, it may not be obvious to the designer how to decompose a problem into manageable subparts. In this case, the designer might then search for an analogous solution based on the high-level problem itself. This retrieved analogical source not only provides a potential solution, *it may also allow the user to infer the problem decomposition in the source design*. This decomposition in the source design (along with solutions to the sub-problems) can be “brought into” the current problem space.

Each new node from the source solution decomposition integrated into the problem

space can act as an additional cue for retrieving another set of solution analogues. This process can continue iteratively leading to the incremental development of the problem space. At every stage of this iterative process, the designer can evaluate the partial solutions available and can decide to take further actions. The iterative feedback between these two processes provides a flexible problem solving framework that accounts for the incremental evolution of complex, compound analogical design solutions. Examples of compound analogical design are presented in Appendix B.

In compound analogy the biological source solution influences the final design outcome. Each analogy brought into the problem changes the conceptualization of the problem itself; modifying the problem model considered for subsequent iterations. In developing the process of compound analogy, however, only the end design was considered; this creates the impression that the analogies are implemented directly to generate a solution to an existing problem aspects. The solution thus generated, it is implied, creates new sub-problems to be solved. However, by considering only the final solution there is necessarily a one-to-one correspondence between analogical source and incorporation of the source into the final solution. Considering the evidence from the RaPower case study, it is reasonable to suggest that many such transfers are made from analogical sources to the problem description; only the final problem description and solution were observed and reported upon in the compound analogy process. Considered across an entire design trajectory, compound analogy as reported, may be a secondary effect of the more routine solution-based problem evolution.

2.4 Applications in support of biologically inspired design

In this section I outline the state of the art in biologically inspired design specific

theories and tools. I break down support into two main categories: process and cognitive support for biologically inspired design, and technological support for biologically inspired design. While the systems outlined below support system modeling, indexing and search, current systems do not deeply address design problem modeling or evolution. Design problems are usually represented outside of the supporting system as a part of the design environment, often provided as a design brief or in the form of a requirements list, or both. Problems are either not represented by the system or are expressed as lightweight and usually fixed models at the start of design, thus the freedom to alter and track changes to design problems is not supported. One contribution of this dissertation is that it can be used to build tools to fulfill this need, and a prototype system is provided.

2.4.1 Process and Cognitive Support for Biologically Inspired Design

Several research groups have evaluated biologically inspired design from a cognitive perspective. Linsey, Wood and Markman (2008), Mak and Shu (2008), Helms, Vattam and Goel (2008, 2009), Helms et al (2008), Vattam, Helms and Goel (2007, 2009) report on cognitive studies of biologically inspired design, while Vincent et al. (2006) proposes a normative theory of biologically inspired design based on the TRIZ theory of innovative design. Wilson and Rosen (2007) provide a process for reverse engineering biological systems to abstract strategies that can be later applied to problems, and Singh et al. (2009) provide a set of strategies for transformation that biological systems employ which might be applied to engineering design. Such strategies may be used to increase efficiency, reduce cost, and increase weight savings.

2.4.2 Support Technologies for Biologically Inspired Design

General design support technology has ranged from interactive design tools that

retrieve design drawings (Gross & Do, 1995, Yaner & Goel, 2002) to collaboration across time and space. Following on the growing movement of biologically inspired design, several organizations and research groups are currently pursuing technology agendas for specifically supporting the process of biologically inspired design.

Biomimicry 3.8 Institute's web portal called AskNature (<http://www.asknature.org/>) provides access to an online functionally-indexed database of research articles in biological sciences. The database is situated in the context of a social networking site enabling designers to better connect with biology researchers.

Chiu & Shu (2007) developed an algorithm that enables engineers to peruse large texts for design-relevant biological systems using functions as search and index terms. Their algorithm uses natural-language analysis, word collocation and frequency analysis, to enable the search and retrieval of relevant biological systems in large text volumes by identifying potential biologically meaningful keywords. Their algorithm was shown to provided for the engineer a set of non-obvious synonyms for function words that may be useful in searching for and retrieving relevant biological systems.

Chakrabarti et al (2005) and Sarkar & Chakrabarti (2008) describe a computational tool (IDEA-INSPIRE) for aiding biologically inspired design, using the SAPPhIRE representation schema to enable functional, behavioral and structural search and referencing of biological source systems. Their work demonstrates, at least in the laboratory context (N = 3), that using their tools with a biologically inspired design process versus non-biologically inspired design process increases the ideation effectiveness of designers by on average 165%. The tool provides the ability to the search of a database of 700 biological entries using the terms in the SAPPhIRE models, and to

display a “human understandable” representation of the biological system.

Shu, et al (2007) show the feasibility of producing function basis models for biologically systems, and then provide a case study demonstrating the usefulness of using function basis models for analogical transfer between existing biological and technological systems. Importantly, they note the process of analogy may occur at different levels of functional abstraction. Nagel, Stone and McAdams (2010a), further extend the concept of abstraction to include both category and scale abstractions. Cheong, et al. (2011) provide a basis of “biologically meaningful keyword and functional terms” which Nagel, Stone and McAdams (2010b) further extend by providing a “thesaurus” that enables designers and biologists to translate standardized functional basis terminology into biologically equivalent terminology and vice versa, for the earlier developed “meaningful keywords and functional terms.” This further ameliorates the indexing and search problem between the disparate domains.

Vattam, et al (2010) developed a tool, DANE, based on Structure-Behavior-Function (SBF) models and the cognitive models developed in Helms, Vattam, and Goel (2008, 2009). This tool provides designers with both the capability to construct SBF models of biological and technological systems, as well as to search and browse a library of such tools using functional keywords, functional relationships (graph navigation), and keyword search. This tool has been implemented in the context of a biologically inspired design class, where it was useful for generating useful discussions among student teams on the challenges of functional naming, indexing and retrieval, and in the case of one design team, proved useful for helping students structure their design thoughts in terms of abstract functions.

Vattam and Goel (2011) have likewise developed a model based tool for indexing and retrieval of relevant documents associated with biological systems that may be relevant for a design case. Such indexing and retrieval is a common yet challenging task in the classroom environment, where students are required to retrieve many such supporting documents to further their understanding of the biological systems to be transferred.

In the systems developed by Nagel, Stone and McAdams and those developed by Vattam, Wiltgen, Helms and Goel, designers may directly inspect functional models of biological systems. Such systems use these models to not only index and search for relevant biological sources, but also to transfer useful design concepts from biology to the design context. Such models benefit from structural independence, enabling engineers to transfer functional models and then implement with alternative structures more amenable to human manufacture and with performance characteristics specific to the design problem at hand.

These systems above represent the state of the art in technological support for biologically inspired design. The research results for the tools so far developed are focused primarily on indexing, retrieval and transfer of analogies for design; that is, given some target problem, how does one find and transfer the best analogical source system to help solve the design problem. Despite this commonality, one challenge of evaluating these tools is that each uses a different set of design problems, is implemented in a different context, and uses different criteria to evaluate the performance of each system.

3 CONTEXT AND METHODOLOGY

In this section, I discuss the context of study both in terms of the newly evolving discipline of BID, and more specifically in terms of the interdisciplinary class on biologically inspired design in which this research takes place. I will follow the discussion of context, with a discussion of the overall case study research design used in this dissertation.

3.1 Biologically Inspired Design

Biologically inspired design (hereafter referred to as BID) is an important and growing movement in design (Goel, McAdams, & Stone 2013; Shu, et al 2011; Bar-Cohen 2011, Bonser & Vincent 2007; Yen & Weissburg 2007, Vincent & Mann, 2002; Benyus, 1997). The movement is driven in part by the need for environmentally sustainable development, and partly by the recognition that nature can be a powerful source of inspiration for technological innovations. Common examples of biologically inspired design include: fasteners (Velcro) inspired by the burr plant, dirt repellent paint inspired by the lotus plant, more efficient and quieter wind turbine blades inspired the fins of whales, etc.

Biologically inspired design has a rich tradition in design, dating back at least as far as DaVinci where his attempts to conquer flight provide an excellent study in the practice. In attempting to conquer flight, DaVinci began first with the design of a parachute, based on the simple geometry of a pyramid. This was followed with a more ambitious design of a machine, the “screw air”, based on the mechanics of the screw and the predecessor to today’s modern helicopter. Later, DaVinci turned to imitating nature,

developing winged device meant to attain flight through a flapping motion powered directly by a man. Finally, DaVinci developed a glider, meant to attain flight by leveraging power to be found in the air itself (Bartoli, et al 2009).

This series of progressive designs forms a interesting, even representative, design pattern; first in the case of a pyramid shaped parachute a simple design testing fundamental principles of shapes and behavior; second the application of known engineering principles, adapting another known design, the screw, to a new domain; third, the superficial mimicry of a system known to produce the desired behavior, in this case the flapping wings of a bird to produce flight; and finally, through deeper insights of the mechanism of bird flight, and the adaptation of both the problem and the design to arrive at a suitable design. A brief read through “The Codex on the Flight of Birds” demonstrates DaVinci’s gift for developing these deep insights through observation. Later, similar observations of birds, in particular shape changes in wings, provided critical insight to the Wright brothers in developing successful control of roll during flight. Even today, modern aeronautics still looks to birds as models of control and maneuverability in flight. While biologically inspired design has been in practice for centuries, only now, in the past two decades, are we recognizing this as design practice in its own right and exploring it as a science of design.

As a modern trend, Bonser (2006) projects that as of 2005 we are a little more than half-way through a 40 year innovative growth cycle in biologically inspired design (assuming a sigmoid growth pattern, common to information diffusion models). In a follow up study I conducted in 2013 (see Appendix A), using the same method as Bonser from 2005, but extending the data through 2012, we see a ten-fold projected increase in

number of patents produced through biologically inspired design, with the trend of innovation lasting at least through 2040. Consider further that according to the Encyclopedia of Life (www.eol.org), experts estimate that “there are at least four times more complex species alive on our planet as the 1.9 million that have already been discovered and named, and that “of the species named, only a fraction have been studied in nature beyond their initial naming.” We must be open to the possibility that the upper bound on the biologically inspired trend is in fact significantly greater than even the new trending model projects.

In addition to the raw potential of the field to impact design and innovation, I believe biologically inspired design merits study by cognitive scientists for at least three reasons. Firstly, because biologically inspired design is fundamentally analogical, it provides ideal conditions for studying analogical design. In other design methods, analogy may be one tool among many, to be used opportunistically, or as chance provides. Indeed many cases of so-called biologically inspired indeed appear to be serendipitous, including the famously described discovery of Velcro. This oft-cited discovery is certainly an example of biologically inspired design. The discovery of Velcro, however, was not the product of a deliberate and systematic approach to leverage biology. As a domain of study for cognitive scientists, biologically inspired design is an environment that exists “naturally” outside of the labs of cognitive scientists, that provides an extremely rich environment for the study of the process of analogical design. Because such biological sources are radically different from the domain to which they are applied, the process and products from these analogies are more easily observed. Often multiple analogies are seen over the course of an entire design trajectory (Helms, Vattam, & Goel 2008, Helms, Vattam, &

Goel 2009). Such prolific and easily identifiable use of analogy makes biologically inspired design an ideal domain for the study of the underlying processes and mechanisms of design by analogy.

Secondly, biologically inspired design requires analogy-making across a wide disciplinary gap. Whether the design domain is industrial, chemical, materials, mechanical or aeronautical engineering, architecture, computer science, or industrial design, the leap from that domain to biology is wide. Studies of interdisciplinary design from the emerging and important context of design between electrical and mechanical engineers, for example, involve crossing gaps of specific domain knowledge between one domain and another, but design goals, process, training and values (ethics) of the two disciplines remain very similar. Thus the framework of design within which the discourse occurs provides many shared points of reference. This is distinctly not so for biologically inspired design. The context in which a biologist operates differs as much in goals, process, and value (ethics), as it does in fundamental domain knowledge. The context difference is so profound that even the meaning of fundamental concepts like “function” present in both fields, becomes difficult to disentangle. Thus biologically inspired design provides a high contrast context for understanding interdisciplinary design.

Thirdly, biologically inspired design takes place in fundamentally innovative contexts. The results of most cases of biologically inspired design I have studied or witnessed are original designs, in the sense that they are neither adaptive (the application of an existing system to a new task) nor variant (a change to size, quantity or arrangement of an existing design) designs (Gero, 2002; Pahl & Bietz, 2003). Original design requires the application of a new solution principle to solve a design problem. In this sense,

almost all biologically inspired designs are original⁶. Thus biologically inspired design provides a context for studying design where the design goal is usually innovative or creative design, and constraints are relaxed (the degree to which depends on the context). The context of all instances of design in this dissertation, which takes place over seven years in the context of a biologically inspired design classroom, involves unconstrained design with explicitly stated goals (and rewards) for innovation and creativity. Biologically inspired design fosters a research context that enables innovation, something highly prized in design.

3.2 The Biologically Inspired Design Class

The theories developed in this dissertation are situated in the context of a course on biologically inspired design taught at the Georgia Institute of Technology. This section will cover the details of the course, and its evolution over time.

The rapid growth and interest in the field of biologically inspired design is driving the development of educational courses for supporting biologically inspired design in practice. Georgia Tech's Center for Biologically Inspired Design (<http://www.cbid.gatech.edu/>), offers a senior-level interdisciplinary course on biologically inspired design. According to Yen, et al (2011), "The connection between engineering and biology provided by BID as a problem solving activity provides an excellent atmosphere in which to encourage interdisciplinarity and develop sound pedagogical practices." The course defines five key learning goals: 1. Novel design techniques, 2. Interdisciplinary communication, 3. Science and engineering knowledge

⁶ Exceptional instances do occur where biologically inspired design techniques do lead to the "re-discovery" of an existing solution – for example the discovery and use of fiber composites, in the case of a team examining tree root structures to use in building tunnel supports [BID class final report, circa 2007].

outside core domain, 4. Interdisciplinary collaboration, and 5. Application of existing knowledge to a new field (Yen et al, 2011). It is grounded in the theory and practice of interdisciplinary research and education, recommended in the cognitive and learning sciences (e.g. Ausubel 2000; Bransford, Brown & Cocking 2000; Bybee 1997; Lave & Wenger 1991; Vygotsky 1978) as well as recommendations for teaching science (e.g., National Research Council 2011) and biology (e.g., National Research Council 2009).

In 2006 and 2007 I conducted in situ cognitive studies of design teams in the biologically inspired design class. These studies have already led to the advancement of new cognitive frameworks for biologically inspired design, and have influenced how the class is structured over the last five years. For instance, problem-driven and solution-based design processes (Helms, et al 2008) are now taught explicitly in class, and at least one of each type of project is required for each design team. Compound analogy (Helms, Vattam, & Goel 2008) is also an explicitly taught technique and the organizing framework of structure-behavior-function (Bhatta & Goel 1996) is embedded throughout classroom exercises, homework and design reports. These techniques and their justification are fully described in Yen et al 2011, and Yen et al 2014.

3.2.1 Course Content and Assignments

Although many elements of the course changed throughout the years, the elements that remained constant include content lectures, found object exercises, and one or more interdisciplinary team-based design projects. Lectures are focused on (a) exposing student designers to specific case studies in BID, (b) exposing student designers to design processes and representation techniques specific to biologically inspired design, and (c) providing designers with specific technical skills to perform design analysis, such as

quantitative engineering and materials analysis. The lectures are provided by a combination of course instructors and guest lecturers with experience in the field.

Found object exercises require designers to bring in biological samples and to analyze the solutions employed by these samples. Students are encouraged to perform background research by providing at least two scholarly articles on each object, and to perform first-hand experiments using their objects to determine the useful properties and functions. For instance a student may expose a closed pine cone to a humid environment or directly to water and measure the rate at which the pine cone opens. After the first found object exercise, which is open to any object the student cares to use, further found object exercises are focused on particular properties or functions such as locomotion, sensing, or hierarchical materials. These found object exercises are usually paired with a corresponding BID case study lecture, for instance a found object exercise on locomotion may be due during the week of a lecture on the locomotion of organisms and robots that involve interaction of matter with complex media complex fluids or granular media (e.g. lizards walking in sand or cockroaches walking over bark and leaves) (DanGoldmanPaper). There are between four and six found object exercises in each year.

The design projects start by grouping an interdisciplinary team of 4-6 students together. Instructors ensure that each team has at least one designer with a biology background and a few from different engineering disciplines. In some class iterations, teams were allowed to select any design problem, in others, the class was focused around a more general design domain⁷ within which teams had to find a problem to work on. In some years teams were assembled by the instructors based student self-reported topics of

⁷ In 2008 the design domain was addressing solutions for the global supply of potable water; in 2009 the design domain was design for adaptable, sustainable housing.

interest, in other years team assignment was more arbitrary⁸. For all projects, each team was to research their problem and design a solution using biologically inspired design, based on one or more biological systems. While the number and timing of design projects varied by year (from between one to three projects), all teams presented design concepts at least once during the middle of the term, and then submitted final designs during the last two weeks of class along with a final design report.

3.2.2 Course Instructors

The biologically inspired design class at Georgia Institute of Technology, is team-taught by an interdisciplinary group of instructors. Several core instructors organize and lead the class, while many guest lecturers providing content lectures throughout. The composition of core instructors and guest lecturers varied year by year. In all years, Jeannette Yen, Professor in the school of Biology and Director of the Center for Biologically Inspired Design (CBID) was the primary instructor, designer and organizing “force of nature” for the class. Jeannette received her PhD in oceanography, and performs research on biological oceanography and zooplankton ecology.

Prof. Marc Weissburg in the school of Biology and Co-Director of CBID provided support in the first 2 years of class, and served as a guest lecturer thereafter. Prof. Weissburg received his PhD in ecology and evolutionary biology and studies “the mechanisms of information acquisition for fluid mechanical and chemical signals by animals, and the consequences of perceptual abilities for populations and communities.” (<http://www.cbid.gatech.edu/directors.html>)

Prof. Craig Tovey from the School of Industrial and Systems Engineering and Co-

⁸ The exact process of team assignment varied by year and was not available for inspection.

Director of CBID has likewise been a part of the core instructor team. Prof. Tovey received his PhD in operations research, and performs research on “inverse optimization for electric grid management, classical and biomimetic algorithms for robots and webhosting, the behavior of animal groups, sustainability measurement, and political polarization.” (<http://www2.isye.gatech.edu/~ctovey/>)

Prof. Ashok Goel from the School of Interactive Computing, Co-Director of CBID, and Director of the Design and Intelligence Laboratory (DILab), in addition to providing guest lectures on design representation and process has also played key role in the design of the class. Prof. Goel performs research on cognitive theories of biologically inspired design and design learning as well as artificial intelligence technologies for supporting the practice and learning of biologically inspired design. The theories of biologically inspired design developed in DILab, such as those mentioned earlier: Compound Analogy, Problem-driven and Solution-based Design, and Structure-Behavior-Function theory, provide the underlying theoretical scaffolding for many aspects of the course design.

Table 3-1 provides a (partial) list of guest lecturers in the course. In addition to lectures, each team was also provided one or more faculty as mentors to provide expert advice when needed. In some years, project teams met with mentors independently outside of class, while in some years mentors were invited into the classroom to provide advice during working design sessions. Design team engagement with mentors varied by team and year.

Table 3-1. Partial list of professors and lecturers for the biologically inspired design class at Georgia Institute of Technology, 2006-2012.

Lecturer	Affiliation
Bert Bras	School of Mechanical Engineering, Georgia Tech
Charlie Camarda	Senior Advisor for Innovation, NASA
Young-Hui Chang	School of Applied Physiology, Georgia Tech
Mehmet Dogu	Research Knowledge Manager, Perkins & Will
Dan Goldman	School of Physics, Georgia Tech
Connie Hensler	Director of Lifecycle Assessments, Interface Carpets
David Hu	School of Mechanical Engineering, Georgia Tech
David Ku	School of Mechanical Engineering, Georgia Tech
Jason Nadler	GTRI, Electro-optical systems laboratory
David Oakey	Founder, David Oaky Designs
John Reap	School of Mechanical Engineering, Georgia Tech
Jim Spain	School of Civil and Env. Engineering, Georgia Tech
Mohan Srinivasarao	School of Polymer, Textile and Fiber Engineering, Georgia Tech
Julian Vincent	Department of Mechanical Engineering, University of Bath
Steven Vogel	Department of Biology, Duke University
Bruce Walker	School of Psychology, Georgia Tech
David Webster	School of Civil and Env. Engineering, Georgia Tech
Claudia Winegarten	Industrial Design Program, Georgia Tech

3.2.3 Student Composition

Students in the course were self-selected from a population of interdisciplinary

Georgia Institute of Technology undergraduate and graduate students. The course recruits students from these majors: mechanical engineering, industrial and systems engineering, materials science engineering, biomedical engineering, and biology; though the course is open to any major and they sometimes get majors from industrial design, architecture, chemistry, mathematics, or nuclear engineering. The course is restricted to juniors and seniors who have established their majors, thus are able to bring specialized knowledge to the design process. As documented in Yen et al (2011) in 2006 and 2007 the ratio of biology majors to engineers (and others) was approximately 1:4, which shifted to a ratio of 3:5 in subsequent years. The ratio of biologists to other majors was accomplished by shifting the allocation of available slots for a given major during registration. In the initial registration process, students are able to register on a first-come-first-served basis for a course only so long as slots for their major remain open. After all students have had the opportunity to register for their major, students are able to register for unfilled slots in other majors. In this way, architecture or industrial design majors, for example, may be able to register for unfilled slots for materials science engineers. Even after registration, during the first weeks of class, students are able to petition the instructor for an exception to gain entry into the class. The number of students and process by which the instructor accepted students who petition is unknown.

One important observation about the students entering the class is that biologists have no required design process training, and undergraduate engineers have little formal training, although they tend to have more experience. Furthermore, engineering students' experience before their senior design project tends to be with closed design problems, where answers involve the application of well-studied principles to well-understood

problems. In contrast, BID problems tend to be open ended and ill understood at the beginning (Yen et al, 2011).

3.2.4 Biologically Inspired Class Design Example: RaPower, 2009

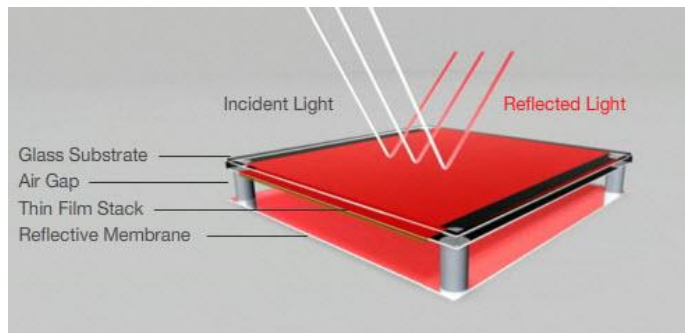


Figure 3-1 Final design diagram of SolShield from the RaPower design team.

The following provides one example of biologically inspired design from the 2009 class. The team was formed by the instructors based on feedback from student preferences, and consisted of one biologist, one mechanical engineer, one industrial engineer, and one chemical engineer, and was provided with a focus on the problem of energy in the context of sustainable housing. This focus led eventually to a biologically inspired color changing cover for solar thermal water heaters to prevent overheating. The description of the design trajectory provided here is based on the analysis of four homework assignments turned in sequentially over the course of the semester approximately 2 weeks apart: a description of the problem; a midterm presentation; a second problem description; and the final presentation.

The team began with the initial open-ended problem of sustainably generating power. After an initial meeting, the team produced a range of sustainable types of energy – wind,

solar, water, geothermal – discussing solutions such as wind turbines, photovoltaic cells, towers of liquid sodium heated through reflected light, chemical batteries, and storage of energy for later use using compressed air. The design team also mentioned fat as a means of storing energy in biology. Cost was highlighted as a salient constraint on their design. The designers also ranged extensively in discussing different places in which the current technologies were used: from coastal areas, to farms and cities; they also discussed relevant weather conditions, such as the amount of wind or sun, and extreme conditions such as those found in Death Valley. Criteria were vague, of the character “more efficient” or “costs less”.

The midterm presentation limited discussion of existing technological solutions to photovoltaic cells and coal plants, however a wide range of biological sources were considered including the desert snail, diatoms, photosynthesis, enzyme reactions, and the lotus leaf. High-level descriptions of the relevant functions of each biological source were described, for example that the function of the desert snail is heat dissipation, performed by the structure of its shell. The designers proceeded with proposing simple one-to-one corresponding solution-modifications to the photovoltaic cell, derived from each of these biological solutions. Thus, in the case of the self-cleaning lotus leaf, students proposed a self-cleaning photovoltaic cell. Solution proposals were little deeper than a function-solution pairing of the type just mentioned, none of which were developed further. From initial description to midterm, we notice the addition of new functions, cleaning-self and dissipating heat which were directly associated with biological solutions having the same functions. I note that designers drop other heat related functions, such as storing and directing heat. I also note that while the mirror/heat-

tower solution is dropped, the environment in which it operates, the desert, remains in place, and is also the same environment in which the desert snail lives. Furthermore, the criteria “passively” is now associated with the heat dissipation and self-cleaning functions that were attached to biological solutions. Manufacturing also is a rising concern, as the ability to reproduce materials and effects is highlighted.

The second problem description assignment continues its focus on solar panels and photovoltaic cells. All of the biological sources mentioned previously, are maintained, except diatoms which appear to have been dropped. Heat dissipation is discussed, but the design team now focuses on a flexible, moldable and self-cleaning surface derived from the lotus leaf, and on a newfound perceived deficiency in current solar panels – rigidity. Furthermore, the operational environment has shifted from a desert focus, to a more dynamic environment with greater temperature range. As well, the team is focused on the need to connect their solution to a home (part of the initial design requirement). Again, students raise manufacturing of nanoscale materials as a concern, as well as the need for materials to be sustainable. The criteria focus has shifted from passive response in the midterm presentation to increased efficiency.

Figure 3-1 provides a graphic of the final design. Students arrive at a new solution, which is concerned with regulation and cooling, rather than self-cleaning and flexibility. The design team appears to have radically evolved the problem, now no longer working with photovoltaic cells, but looking at solar thermal collectors for water heating, which run the risk of overheating and damaging their internal structure. The solution is a dynamic feedback regulation mechanism from enzymes discussed in the midterm, combined with a solution inspired from a new biological organism, the tortoise beetle,

which has a shell which it uses for camouflage by changing color. The designers intend to use a mechanism similar to the tortoise beetle to alter the color of the thermal collectors to change the amount of heat captured, depending on the internal heat of the unit. The final design, the SolShield, is the first solution generated as more than a simple function-solution concept.

While this final problem appears to be a new problem, one can see, in fact, the derivative nature of the process. Reacting to heat has been embedded in the teams thinking all along, from the mirror/heat tower, to the desert snail, to the operational environment of the desert, to the concept of dynamically responding to the environment. These concepts were influenced by a number of previous solutions that were investigated so that when a new problem concept arose – overheating -- the team was able to pivot to the new problem focus and quickly come up with a dramatic, creative solution.

3.2.5 Key Changes in the Classroom Environment 2006-2012

One of the key assumptions of this research is that while each class is unique in terms of students, instructors, and syllabi, the observed phenomena remain constant and are supported by similar cognitive processes over these variations. I provide a complete description of all of the changes that occurred in Yen et al (2014). In this section, I describe those changes to the structure of the class relevant to the work in this dissertation.

3.2.5.1 *Problem vs. Solution based design*

Initial iterations of the class assumed a tradition problem-driven design methodology. In 2008-2009 the concept of solution-driven design was introduced to the course as a process that could be used for student design, although it remained optional and only one

design project was required in those years. In 2010-2012 the course used multiple designs, one design that used the problem-driven process and one design that used the solution-based process. In each of these years, students started with the solution-based design first, and presented a problem-driven design second.

3.2.5.2 *Problem-orientation*

Initial iterations of the class allowed students to define their own design problem, with very few constraints. The constraints that instructors imposed were framed in terms of generating an innovative conceptual design that could be “pitched” to a venture capital firm. In 2008 and 2009 the problem domain was curtailed to a water-related problem (e.g. filtration, distillation, harvesting, etc.) in 2008, and to a sustainable housing related problem (e.g. saving on power use, HVAC systems, reducing water consumption, etc.) in 2009. These constraints were removed in 2010-2012 to once again allow for completely open problem formulation.

3.2.5.3 *Analogical evaluation*

In initial iterations of the class, student designers identified (usually) numerous biological sources and applied a subset of these to develop a solution. How many analogies were made, and how well these translated into a design were the major criteria for evaluation. The process of analogue, in particular of analogical evaluation – why certain analogies were selected while others were not -- was opaque in this context. Analysis of analogues prior to 2009 focused on student designs’ ability to articulate their understanding of the underlying mechanism of the analogue, rather than a justification of the analogy itself. In 2010-2012, specific assignments and final report criteria were added to the course requiring students to *justify* their biological analogues.

3.2.5.4 *SBF Representations*

In 2007 SBF (Bhatta and Goel, 1997; Goel, Rugaber, & Vattan, 2009) was introduced by instructors as a framework for organizing found object exercises. SBF is grounded in cognitive theories of systems thinking, and may be summarized in brief as follows:

- Structure, behavior, and function form an abstraction hierarchy for systems thinking; behavior is an intermediate level of abstraction between structure and function.
- Structure specifies the components of the system as well as the connections among them. For example, the structure of the electrical circuit in an ordinary household flashlight comprises of an electrical battery, a light bulb, a switch, and electrical connections among the battery, bulb and switch.
- Behaviors specify the causal processes occurring in the system. For example, the behavior of the flashlight is that when the switch on the flashlight is pressed, current flows from the battery to the bulb, and the bulb converts electrical into light energy.
- Functions specify the outcomes of the system. For example, the function of the flashlight is to produce light when the switch is pressed.
- Behaviors provide causal mechanistic explanations of how the structure of the system accomplishes its functions. For example, the behavior of the flashlight explains how its structure accomplishes its functions.
- A behavior of a system specifies the composition of the functions of its sub-systems into the system functions. For example, the behavior of the flashlight composes the functions of its components—the battery, bulb, and switch—

into the function of the flashlight.

A subsystem or component of a complex system can itself comprise a system and thus have its own SBF model. Hence, SBF models of a system can have a hierarchical structure. For example, consider the system of the basilisk lizard, which is well known for its ability to run across water. If the function (F) of interest of the basilisk lizard is “run on top of water,” one can consider the opposing limbs, tail, and wide flat feet as part of the structural (S). The way in which the feet move in opposition are counter-balanced by the tail, and how the feet slap the water generating lift, then extend down and back creating more lift, thrust and a pocket of air in the water, and are then withdraw up and out through the air pocket could be considered the behavior (B) that generates the “run on top of water” function. One could consider the muscular-skeletal system of the legs as a subsystem of this system used to create a subfunction “generate movement of legs” which causes the higher-level “run on top of water” function. Likewise, one can consider the form the foot takes throughout the process as another sub-function, “change foot surface area.” In this way one can decompose the “run on top of water” function into a number of sub-functions, including “generate movement of legs” and “change foot surface area,” each of which could entail another SBF model. Similarly, one can consider the function “run on top of water” to be part of the function “escape predator” showing that one can navigate both up and down the levels of functional abstraction in the SBF model hierarchy.

The origin of SBF analysis lies in Chandrasekaran’s functional representation scheme (Chandrasekaran 1994; Chandrasekaran et al. 1993). Other researchers have developed

similar cognitively oriented approaches to thinking about complex systems, for example, Rasmussen (1985). Gero and Kannengeisser (2004) describe the design process itself in terms of function, behavior, and structure. Erden et al.(2008) provide a recent review of functional modeling. Note that in SBF analysis, functions are mental abstractions chosen by the modeler, and not intrinsic to the complex system. In the case of engineering systems, a functional abstraction corresponds to an intended output behavior of a system, subsystem, or component. However, since functions are mental abstractions, we can also use SBF modeling to model natural systems, including biological systems, such as the human heart, and ecological systems, such as forests. Even more so than engineered systems, natural systems exhibit layers of varied functionality at different scales, feedback loops, and other types of causal processes that characterize complex systems

Students were asked as part of the Found Object homework assignments and in their discussions to (a) focus on a single *function* of the organism in question, (b) identify the *structures* relevant to accomplishing that function, and (c) provide a *behavioral* explanation for how those structures give rise to the function. Instructors facilitated these discussions as necessary to guide students. (In the SBF vocabulary, behavior is synonymous with causal explanation or mechanism.) To simplify the vocabulary, in 2008, the SBF vocabulary was changed to a What-Why-How vocabulary, mapping “What” to “Structure”, “Why” to “Function”, and “How” to “Behavior.” This was an attempt to both remove the ambiguous interpretation of “behavior” and to formalize the levels of functional abstraction. Functional abstraction was considered in terms of “why” moving up the hierarchy (more abstract, super-functions), and “how” moving down (more detailed, sub-functions). Again, students were asked to describe all biological

systems in these terms, both conversationally and in formal homework assignments and design reports.

In 2011-2012, the framework was again changed, this time to include the SR.BID framework developed in this dissertation. The conceptual framework of SBF was integrated into SR.BID, and the SBF and WWH vocabularies were dropped from class materials.

3.2.6 Role of the Researcher

My own relationship with the course, instructors and students has varied over time. In 2006 I passively observed the course, attending classroom lectures and reviewing design reports and presentations. In 2007 in addition to being a passive observer, I conducted two short (one-class) experiments in the class. I also worked with Prof. Jeannette Yen in tailoring the biologically inspired design instruction to NASA designers, through NASA's NESC Academy, which was delivered in July 2008.

In Fall 2008 and beyond, my role was more involved both in terms of course design and instruction. As the theories developed by myself and my colleagues in DILab influenced the pedagogy of the class, Prof. Yen kindly invited me to assist in the overall design of the course. In 2008, 2009, and 2010 my involvement included assisting Prof. Yen in course design and in providing one or two guest lectures on design process each year. I also remained present as a passive observer during some of this time, and provided limited feedback to the instructors on observations of conceptual designs submitted by the students. During this time, at the behest of the instructors and with my assistance, SBF representations and organizing frameworks, functional decomposition, and solution-based design were incorporated into the course, and the number of conceptual designs

was increased so that both problem-driven and solution-based design processes could be experienced by the student designers.

In 2007, I became aware that students were spending much of their time working on understanding, formulating, changing and reformulating their design problems. Moreover, observations indicated that the biological sources students observed influenced their problem formulation decisions. My involvement in the course allowed me to present this insight to the instructors and subsequently to advise them on how they might provide support to students. As a consequence of these discussions, the instructors decided to include open-ended, unstructured design problem statements as assignments and in team design reports. These interventions were theory-neutral, in that they made no commitments to the representation or method by which problem⁹ formulation occurred.

By 2011, based on the data we had collected, I had formulated a more structured method of formulating problems. In 2011 and 2012, I worked closely with the instructors to implement several interventions in the class. These interventions were not theoretically neutral. In the first intervention included in 2011, the conceptual framework of Structured Representations for Biologically Inspired Design (henceforth SR.BID) was implemented using the four-box method of solution and problem specification. The framework was used to scaffold the T-chart method of analogical evaluation, also introduced in 2011. These two concepts were direct outcomes of my prior research and were implemented both as proof-in-concept of the tool, and as a means to collect data in a more structured format for further study. In 2012, these two interventions continued, as well as the late introduction of a website structured around team design collaboration using the same

⁹ While problem formulation was theory-neutral, solution formulation was often based on Structure-Behavior-Function theory (later recast as What-Why-How or WWH).

SR.BID schema. In both years, I observed the classes on a daily basis and provided instruction on use of the four-box method, T-chart analogical evaluation method, and the use of the SR.BID website. I also conducted surveys and team interviews during this time.

Thus my role as researcher moved from observer in 2006-2007, to advisor in 2008-2010, to interventionist in 2011-2012. As my role changed with respect to the class, so too did the characterization of the research conducted during these different periods; from observational studies and question formulation, to qualitative encoding and theory building, to rich case study-based evidence building. All claims made in this dissertation will be considered both with respect to these levels of interaction with the class, and with respect to the dynamic and evolving nature of the research (classroom) environment itself.

3.3 Data and Data Collection

A number of different methods were employed to collect data from the context of study for this dissertation. This section enumerates the type of data and the methods used to collect data throughout the seven year course of study. One or more of each of the following sources was used in each study.

3.3.1 Student Demographics

For all students registered in the class, gender, major and year in school were made available. Additional data about students, including academic design experience, design experience outside of the university, motivation for taking the class, etc. may have been collected as part of individual survey protocols (as described in each protocol). Membership in design projects is also known for each student.

3.3.2 Classroom Content

During observational studies of the class core instructors and guest lecturers provided content-based lectures about (a) BID case studies, (b) design processes and representations, and (c) engineering specific applications to design. The speaker and content lectures are indexed via reference to the course syllabus (see Figure X), which lists the topic covered, the speaker and the date of the lecture. This provides an index into the content provided, relative to the time in the design cycle and relative to collected student artifacts. For example, taking the entry “09/20/12 Content 2: Bio inspired sensors Prof.Don Webster” provides an index into the lecture on plume sensing inspired by the blue crab, presented on September 20, 2012. Note, as the course is very tightly scheduled, and heavily dependent on guest lectures, these lectures are scheduled well in advance of the class. There are no known deviations between the scheduled content lectures and the lecture delivered. Specific lecture content (e.g. slides and lecture notes) was not retained for these studies.

Week #	Exercise Tuesdays	Content Thursdays	Homework Assignments (Due noon Tuesdays)
Solution Based Design			
W1	8/21/12: Introduction to Biologically Inspired Design.	8/23/12: Evolution as designer. Found Obj. Primer	8/23/12: Recommended reading: Vogel ch. 1,2, Fish: Mech E article, Helms et al. 2009 [see T2] Assignment 1: Individual: Found Biological Object 1
W2	8/28/12: Review Found B. Object + Investigate nature: search strategies.	8/30/12: Solution Based Design, Analogy Making + Discuss overall project activities.	8/30/12 Recommended Reading: Vogel ch. 12, 14 [see T^2] Assgn.2: Indv.: Solution driven Problem Definition + Analogy
W3	9/4/12 Problems and Problems review. Problem choice for design 1.	9/6/12 Structured Representation for solutions + examples of complete assignment.	9/6/12 Recommended Reading: in biolocomotion (posted on T2), Vogel Ch. 8,9,10. Assgn.3: Indv: Found B. Object 2 (focus on locomotion) Team: Quantitative Analysis 1
W4	9/11/12 Review Found B. Object + QA.	9/13/12 Content 1: Bio inspired locomotion Prof. Young Hui Chang, Applied Physiology	9/13/12 Recommended Reading: Vogel Ch. 6 + Sensing/Signaling readings: posted on T2 Assgn.4: Indv: Found B. Object 3 (focus on sensing)
W5	9/18/12 Review Found B. Object + Team Design.	9/20/12 Content 2: Bio inspired sensors Prof. Donald Webster, CEE	9/20/12 Recommended Reading: Vogel ch. 13 Assgn.5: Team: Design Draft + Presentation.

Figure 3-2. Sample syllabus from 2012 biologically inspired design class.

Additional reading supplied to students in the form of academic papers, or book chapters, is also specified in the syllabus e.g. “recommended reading, Vogel Ch 13”, (Vogel, 2000). In some years the reading was mandatory, and followed by a quiz, in other years, simply recommended. Relevant papers supplied to students may also be found in the resources tab in T-square (online class collaboration and management software), along with their upload date. This combination of records provides an index of content topics delivered to the students and on what date from 2006 through 2012.

3.3.3 Student Assignments

Student assignment data represents the majority of the data used in these studies. From the syllabus and the electronic classroom repository (called T-square) the exact wording and instruction for each assignment is available. Found object assignments were common throughout all years, and were always individual (not team) assignments.

Specific instructions on found object assignments changed from year to year. Many assignments included requirements for finding and including academic references and reference documents, which were also retained. In addition to found object assignments and design project data (covered in the next section), four special classes of assignment data are outlined here. Documentation of these four types of assignments and their variations was collected through regular (weekly or monthly) meetings with the core instructor Jeannette Yen during the summer and fall semesters of 2008 – 2012.

3.3.3.1 *Problem definition assignments*

In 2008 through 2010, additional emphasis was added on design problem formulation. Design problem formulation assignments took three forms. The first form was individual articulation of a design problem. In this exercise, students were asked to provide a brief description of a design problem. In 2010, the first individual assignment on problem definition was given in the third week of class (note, this was in the context of solution-based design) and read:

“Define a new or existing human problem that you think your organism can solve. Use search strategies to find one or more references on existing solutions to that problem, or for brand-new problems on technologies that could be used to solve that problem. Describe the current challenges with the existing solutions/technologies?”

This was followed up in week four with a team problem elaboration phase:

“As a team, perform problem decompositions on the selected natural system and the defined human challenge, making as making links as possible. Iterate between solution (natural system) and technology to identify the key functions necessary in order to translate the biological system to the engineered design.”

And finally for this design iteration a presentation of the design problem in the

context of the overall design during a design charrette:

“For first midterm critique, present your design, show what organisms you started with and which ones you used in your design, using WWH explanations and analogical reasoning for your decisions. Show the problems you considered for your organism, and provide a detailed explanation of which problem your design will solve. Why did you choose that problem over the others? Each team will have 12 minutes to present their design, problem definition and their best biological analogies. Each member must be able to demonstrate a deep understanding of natural systems explored and the problem.”

Variations on the wording and timing of these assignments occurred in 2008, 2009, and 2010. The design problem assignment focuses student attention on early identification and understanding of the design problem, requiring them to more deeply consider the problem prior to generating solutions (as they are often quick to do). The progression from individual, to team, to contextualized problem definition provided an opportunity for progressive deepening of student problem understanding, without imposing constraints. From 2008 through 2010 these assignments were theory-neutral in that they contain no commitments to theories of *problem representation*. While the problem assignments were theory-neutral, the instructors had adopted some *solution* representations prior to the inclusion of these assignments, some of which were influenced by SBF. For example, in the previously quoted instructions, the term “using WWH explanations” stands for What-Why-How, is an adaptation from SBF (see Yen 2014 for details). In this sense, the instructors made their own commitments to a solution representation, which they decided to perpetuate into the problem definition assignment, but only with respect to biological *solutions*.

3.3.3.2 *Four-box assignments*

In 2011 and 2012, based on the theories of problem representation developed in this

dissertation, students were taught the four-box method of problem definition. For all problem definition assignments, except the first assignment, students were asked to use the four-box method of problem definition. The first problem-definition assignment (again a solution-based problem definition) given in week two in both 2011 and 2012 was again meant to be theoretically neutral, and was worded as follows:

“Now that you have a solution and a function defined, and you have some insight into how the biological solution works, think about what kind of problem you can solve using it. Ideally, the problem should be small, tractable, and something that you could prototype or implement as a senior design project. Write a succinct one or two paragraph description of the design problem you are trying to solve. You should NOT write about how you intend on solving the problem just yet. Focus on the design problem you are trying to solve and what makes it problematic. There must be good reasons why someone hasn’t built a better solution already, right? Consider the existing solutions to your problem and what makes them good or bad. You need to think deeply about the problem you are trying to solve and demonstrate that you understand the problem.”

In week seven, I provided students with an instructional lecture for problem definition. This lecture provides theory content and methodology for defining a problem called the four-box method. This method was derived from the theories presented in this dissertation. Figure 3-3 provides an instructional representation of the four-box method.

Operational Environment	Functions
Specifications	Performance Criteria

Figure 3-3. The four-box method of design problem formulation.

In addition to the lecture, an individual problem-definition assignment using the four-box method is given in week seven. The following excerpt from the instructions highlight the differences [emphasis added] between the four-box method and the theoretically neutral approach:

*Now that you have a problem in mind (we'll call this the main problem), use the Structured Representation for Problems that was covered in lecture to write minimum one-page description of the design problem you are trying to solve. After you write up your problem, follow the four box process outlined in lecture to define your problem in terms of **(1) environment, (2) function, (3) specifications/constraints, and (4) performance criteria and why each is important.***

3.3.3.3 Analogical evaluation assignments

Much as in 2008 problem definition was identified as a key course element requiring additional attention in the course structure, so too it became apparent that scaffolding was required for analogical evaluation. Guidance for what constituted a valid or useful analogy in the context of design relied largely on the idea of function matching – if the problem required function X and the biological organism performed function X, it was a match. Issues of material and scale matching were mentioned, however there were no assignments, guidelines or rubrics in place to help either students or instructors evaluate the fitness of an analogy. As with problem definition, in 2008 assignments were added to focus student attention on the process of analogical evaluation.

Again, citing 2010 assignment instructions, in week three students were given the following assignment:

In class we described how analogies can be used to solve problems. To

make an effective analogy, some source (e.g. biological system) needs to be mapped to some target (e.g. your problem specifications). Using the deep understanding of the biological systems obtained from 2 KEY pdf.s and your deep understanding of the limitations of existing solutions to the problem that you identified obtained from 2 KEY pdf.s, construct an effective analogy. Briefly describe your motivation for selecting that analogy: Why is the source relevant to the problem? What are the analogical mappings to your problem specifications? Where are the sources and target different? Provide a rationale for why you selected this design and its proposed benefits over existing solutions.

We note that here, as in the initial problem scaffolding assignments, other than making basic commitments to the process of analogy making – mapping from a source to a target – the assignment remains theoretically neutral in terms of representation and evaluation processes. As with the problem definition, similar assignments are provided at the team and design levels.

3.3.3.4 *T-Chart assignments*

In year 2011 and 2012, we formalized the analogical evaluation assignment using the SR.BID schema and the four-box method of problem formulation. In week three of class, along with defining individual problems (as described above) students are asked to complete an individual analogical evaluation assignment between their problem and a biological source. They are asked to build “t-table” representations but with little conceptual scaffolding (we add scaffolding in subsequent assignments). The wording of the assignment is as follows:

“You have a biological solution in mind, and a problem at hand. Why do you think the solution fits the problem? What are the commonalities between the biological solution and the problem it is attempting to solve, and the design problem you identified in Part 2. Make a T table, list on one side what you know about the biological solution and on the other what you know about the problem. Note where they are alike, and where they are different.

“Now look deeper. For each item on your list, start asking questions: how, what, when, where, why, who, how much, what size, how often, etc. Add these new insights to your T table, and note whether the answers to these questions bring out (1) similarities or (2) differences or (3) there is no corresponding match.

“Construct an argument for why this biological solution is a good analogy to your problem.

“Finally, construct an argument for why this biological solution is a bad analogy to your problem.”

One of the key challenges highlighted as part of why students struggle with analogical evaluation, is that students tend to emphasize the positive aspects of the match, while ignoring those elements that do not match (but might be important). It is for this reason that we ask for the construction of arguments both for and against the analogy in question. Students are asked to do this both individually, and as a team for their first (solution-based) design project.

In exercises for the next two projects, we use the SR.BID schema in the form of four-box models, to represent both design problems (e.g. elevating water, increase traction on icy surfaces, reduce noise from a fan blade) and biological systems (e.g. trees, polar bear feet, owl wings, etc.). Representing design problem and biological system in this way enables a comparison between the two using the common conceptual categories found in the four-box models. Figure 3-4 is used to represent this method to the student designers during training. On the left, we show the design problem represented as a four-box model, on the right we show the biological system, and in the middle column we ask students to provide a qualitative comparison (different, similar, same or not applicable).

Design Problem		Biological System (tree)
Operational Environment		Operational Environment
Individuals and families	-na-	--
Maintenance staff	-na-	--
High-rise building	Different	Streets, parks, forests
Atlanta, urban	Similar	Georgia, ubiquitous
Temperatures 30-100F	Same	Temperatures 30-100F
Functions		Functions
Move water	Same	Move water
Distribute water to units	Same	Distribute water to leaves
Prevent contamination	Different	Transport nutrients
Store water at height	Different	Evaporate water to air
Monitor stored water	Different	Ensure water for respiration
Specifications		Specifications
Non-toxic materials	Same	Non-toxic materials
Non-soluble materials	Same	Non-soluble materials
Installation timing (retro-fit)	Different	Construct while growing
Installation cost	Similar	Cost in energy & materials
Re-use of materials	Similar	Local materials
No external additions	Similar	Internal structures (xylem)
Roof-top storage	Different	Distributed storage
Performance Criteria		Performance Criteria
Height 100m	Similar	Height 30m
Volume 100000 liters/day	Different	100 liters/day
Reduce energy consumption	Similar	Ambient energy use
Water quality standards	Different	Changes to solute concentration
Water flow/timing	Different	Dependent on weather
Waste water	Different	90% lost to evaporation

Figure 3-4. The T-chart method of analogical evaluation. Designers compare a design problem to a biological system across the four-dimensions of the four-box method, using a simple comparison: different, similar, or same.

For the second design, students conduct as a team the analogical evaluation in their second design report. Within the design report instructions, they are provided the following instructions under the Analogy heading in the design report.

“As before, construct an analogy between your problem and you biological source. The table below provides a good example of using SR.BID to build an analogy between a tree and the problem of moving water to the top of a high rise building in Atlanta. For each biologically solution, construct an argument for why it is a good analogy (note the “same” in the T-table) to your problem. Then, construct an argument for it is a bad analogy (note the “different” in the T-table) to your problem. Consider in particular the scale at which the analogies occur, and make sure to highlight any similarities and differences with respect to scale. Likewise for materials.”

Students are asked to perform an analogical analysis for their third design iteration, which is usually (with one exception) an iteration on either their first (solution-based) or second (problem-based) design. The instructions under the Analogy section of the final design are more open-ended and read:

“As before, construct an analogy between your problem and you biological source. Make sure to provide a narrative description in addition to any charts or tables you care to use.”

3.3.4 Student Design Projects

Student design project documentation constitutes another major source of data. The documentation is of three forms: interim design assignments, design reports and design presentations. The number of design projects varied by year, as did type of documentation provided for each project. Here I provide a summary of the (a) project types and schedule by year, (b) type of material available and (c) schedule of material collection by year.

3.3.4.1 Project types and schedule

Projects requirements changed over the years, but can be classified as one of five types: open, constrained, solution-based, problem-based, and third-iteration.

Open: In years 2006 and 2007, design teams were required to work on only one team

project during the semester, with only one significant constraint on the design problem – that the final design must be presentable in a venture capital format (thus it must solve some need in the world, have a viable customer segment, etc.)

Constrained: In years 2008 and 2009, design teams were required to constrain their problem to a domain of interest. In 2008 the domain of interest was “the water crisis,” which led to projects centered on filtration, acquisition, desalination, etc. In 2009 the domain of interest was “dynamic and sustainable housing” which led to projects centered on domestic energy use, heating and cooling, ventilation, water use, etc.

Solution-based: In years 2010, 2011, and 2012 design teams were required to first generate a solution-based design. Design teams quickly decided on a biological system of interest, and identified and designed a solution for a problem that could be solved using that system of interest.

Problem-driven: In years 2010, 2011 and 2012 design teams were required to generate a second design which used the problem-driven design process. Design teams decided on a problem, identified a large number of possible biological systems that might solve that problem, and generated a design solution based on one or more of those biological solutions.

Third-iteration: In years 2010, 2011 and 2012 design teams were allowed to either iterate one of their earlier designs (the solution-based or problem-driven designs) or (in one case) generate a third design using whichever method they preferred.

3.3.4.2 *Project assignments*

A number of smaller assignments related to each project were always scheduled prior to a final presentation and report for each project.

3.3.4.2.1 Problem definition assignments (see above)

3.3.4.2.2 Solution search assignments

These assignments entailed students finding between one and five different organisms to be used as inspiration to solve their design problem.

3.3.4.2.3 Analogical evaluation assignments (see above)

3.3.4.2.4 Environmental impact assessment

These assignments entailed students measuring and evaluating the impact of their final design on the environment, in terms of carbon emissions, toxic materials, energy cost, etc.

3.3.4.2.5 Materials analysis

These assignments entailed students evaluating a number of alternative materials to be used in the construction of their final design, including properties such as strength, toughness, cost, recyclability, manufacturability, etc.

3.3.4.2.6 Quantitative analysis

These assignments entailed students performing a justified quantitative analysis, demonstrating in concept that their design could provide results that would meet customer demand (for example, measuring the theoretical number of gallons of water a fog-harvesting device would collect under a variety of conditions).

3.3.4.3 *Project presentations*

Design presentations were of two kinds (1) full-presentations, and (2) poster presentations. Full presentations entailed the delivery of a slide-show presentation by multiple members of the team, and were typically 20-30 minutes in duration. Poster

presentations were 10-15 in duration, and consisted of a team presentations using single printed poster.

3.3.4.4 *Project reports*

Project reports were of two kinds: (1) full reports, and (2) brochures. Full reports entailed student delivery of a variable length reports usually covering in detail the topics outlined in the assignments section. Brochures were 8-panel, four-page reports that provided abbreviated summaries of the design. They were intended to quickly express the essential aspects of the design (only), and were significantly less detailed than full project reports.

3.3.5 Student Surveys

In 2012, short surveys were completed at the end of the semester by all participants in class. Surveys were given to students in the first 10 minutes of class, while course instructors were not present. The purpose of the surveys was to understand student attitudes about skills required to perform analogy making and evaluation in class, and the use of the four-box and analogical evaluation tools for facilitating those tasks.

3.3.6 Student Interviews & Design Sessions

In 2012 team interviews and design sessions were conducted in the two weeks prior to final design delivery with all design teams in the 2012 BID@GATech class. Initial interviews came at the beginning of a working design session with a class instructor present but non-participating during the interview session. The interviews were intended to review design process information with respect to problem understanding and evolution, and to solicit current design challenges and goals. Interviews followed a high-level script, allowing for follow-up probes as required. From the attending instructor's

perspective, the interviews provided rich background information on the design process, allowing them to provide more targeted advice.

The design interview, which lasted from 10-20 minutes, was followed by a working session design in which the instructor provided additional design mentoring for the student design teams. The working session lasted for the remainder of the hour. Both interview and design sessions were recorded, with the student's permission.

3.3.7 SR.BID for the Web

In 2012, students were introduced to the SR.BID for the Web environment, an online proof-of-concept tool for the use of the four-box and methods in extended, collaborative biologically-inspired design. I provided a 20 minute demonstration of the tool during class, using design data that I previously entered. Data on student use was subsequently collected in the form of transaction summaries by student login ID. Use was voluntary, had no influence on student grades and was largely redundant with the assignments students had been asked to do. See Chapter 8 for a complete description of the SR.BID for the Web environment.

3.4 Research Method

In this section I will describe the mix methods case study methodology used for the dissertation. The purpose of this section is to provide an overview of the high level research design. Within the case study are a number of smaller individual studies, form which the methodology for each will be described later in the section relevant to that study.

This dissertation covers an extended case study from 2006 through 2012 in the

context of the BID@GATech class, which may be broken into in four phases: an *observational* phase, a *content account development* phase, a *process account development* phase, and an *intervention* phase. In each phase, one or more studies were conducted over one of two units of analysis: the individual student designer and the design team (see 3.3.4 for a description of design teams). Each of the studies used one of the following different research methods:

1. *Observational studies* conducted during the observational phase, these studies entail collection of materials and observations of the BID@GATech class in situ, and retrospective analysis through the lens of cognitive theories of analogy.
2. *Exploratory experiments* conducted during the observational phase entailing small, pilot experiments in the BID@GATech class that serve to establish support for further investigation of the phenomenon which are their subject.
3. *Ontologically grounded study* which employs grounded theory or ontologically grounded theory for constructing and validating content theories of biologically inspired design, using artifacts collected in the BID@GATech class; and through the established content theory, enables a mean for systematic qualitative description of the same class of artifacts.
4. *Coded design analysis* is a qualitative study in which an existing coding scheme (SR.BID) is used to analyze artifacts and interviews collected over semester-long design team projects. These studies analyze either snap-shot documentation of design elements at a point in time, the differences in snapshots in design elements over an extended time period, or trace a design

- trajectory over a short design session.
5. *Classroom interventions* in which tools based upon the theories established in this dissertation, are implemented in the context of the BID@GATech classroom for the purpose of understanding the degree to which these technologies are adopted and accurately used by student designers.
 6. *Computational implementation* in which computational applications based upon the theories established in this dissertation are implemented and shown to be computationally feasible.

A more detailed description for each methodology will be discussed in the section corresponding to the relevant study.

3.4.1 Exploratory Phase

In 2006 and 2007 I performed *in situ* observational studies of the BID@GATech class. While there were no research problems per se driving these studies, the purpose of these studies was to (a) provide a descriptive account of the activities occurring in the biologically inspired design classroom, (b) identify and describe those processes used by instructors and students to perform BID, in particular as they differed from standard design processes and theories, and (c) identify those challenges faced by students and designers, which may be both informed and assisted by cognitive and computational sciences. The critical output of the observational phase is a set of research problems.

During the *in situ* study observations took place twice weekly in the context of the BID@GATech course. A second researcher also participated in these studies. A research notebook was maintained for recording observations and field notes. Design reports and presentations were made available to the researchers at the conclusion of the class. For

each of the design teams, design trajectories tracing (a) the conceptual development of the design, and (b) the biological source(s) of inspiration associated with each design were compiled from the periodic presentations and collected works of the students. Case studies in biologically inspired design presented or made available by the instructors (as additional reading) were also collected. The two researchers discussed findings after each session, and reviewed findings periodically with Prof. Ashok Goel, a senior researcher in the field of cognitive science and design.

In addition to observational data gathered, two classroom based exploratory experiments were conducted during this phase. The first of these experiments tested the effectiveness of the different representation types for assisting individual students understand and reason about biological systems. The second experiment tested whether biological source analogies were influential on how the class framed a design problem.

Finally, as part of the exploratory studies, I conducted a coded design analyses over a set of homework assignments collected in 2007 to further understand the connection between solution analogies and problem formulation at a point in time. The methodology and details for each of these experiments and assessments will be described in detail in subsequent sections. For easier reference, exploratory studies use the “E” prefix in their designation, as shown in Table 3-2 which summarizes the exploratory studies.

Table 3-2. Exploratory studies and associated study type.

Study	Exploratory Studies (E)	Type
E.1	2006, 2007 <i>in situ</i> observation of BID classroom	Observational Study
E.2	2007 Test of biological systems on problem decomposition	Exploratory experiment
E.3	2007 Data analysis of design problem decomposition	Coded design analysis

The collective findings from the exploratory phase of this case study provides evidence that (a) students struggle with problem formulation, (b) that biological analogies play a role in the formulation and evolution of design problems over the course of a design trajectory, and (c) that students have difficulty making appropriate analogies. These observational studies form the basis for the first two research problems.

3.4.1.1 *Research Problem 1*

While many theories of design problem representation exist, it is unknown to what extent current content theories of design can support the process of design problem formulation and evolution in BID.

This first problem examines the conceptual categories and relationships entailed in the study of design problems. Existing content accounts of *design problems* in design theory are generally either (a) prescriptive, as in requirements gathering protocols, or (b) very abstract (e.g. functions and constraints). For a study of problem evolution, we require a lens by which the problem may be systematically examined. We require a rich content account of design problems that accounts for the relationship between biological analogies and design problems, for the breadth of biological analogies and design problems observed in the classroom context. This account must be capable of systematically describing the changes occurring in a design problem in such a way that one can correlate such changes with the biological analogy. The developed system, a content account of design problems and problem-solution relationships, must also provide sufficient generalization to account for the variety of biologically systems and design problem observed. These observations lead directly to the first problem addressed in this dissertation.

3.4.1.2 *Research Problem 2*

Current theories of analogy account for analogical evolution of solutions; while current theories of problem evolution account for problem evolution in the context of conjectured solutions. Analogical problem evolution in response to existing analogical solutions remains unaccounted for in design theory.

The second problem follows directly from the claim that problems evolve in response to biological analogies. Current theories of analogical design and problem-solution coevolution exist in the context of a conjectured design solution, which are different from existing design solutions, and even more different from design solutions from distant domains. Theories of analogical design do not address the issue of problem evolution as a first class phenomenon. While theories of problem-solution coevolution provide an explanation for problem evolution in the context of a conjectured solution, no current theory of problem-solution coevolution provides an account for how or why a biological analogy may exert direct influence on the evolution of a design problem in the absence of such a conjectured solution. Thus these observations lead to the second research problem:

As the solution to the second problem is dependent on the solution to the first, I address them in order: first developing a content account, and then a corresponding process account.

3.4.2 Content Account Development Phase

During the second phase of the case study, I explore the first research question posed. I first conduct a literature survey on existing theories of representation of problem formulations. I evaluate existing theories with respect to their capability for supporting

the task of analogical problem evolution in biologically inspired design. From this evaluation I determine that SBF and SAPPPhIRE representation schemas provides the most capability with respect to the task. I select SBF for follow up due to familiarity relative to the alternative.

Starting with SBF, I derive a content account called SR.BID from data about biologically inspired design. The selection of SBF is predicated upon three data points: the first is the analysis of other theories of problem representations, the second is an exploratory experiment conducted in 2007 documenting the utility of SBF to aid in question answer in the BID@GATech class context, and the third is the documented prior use of the SBF ontology for describing complex biological systems in the context of the BID@GATech class. I use a variation on the methodology of grounded theory [Glaser & Strauss 1967; Strauss & Corbin 1990]. In the grounded theory methodology, a theory about any phenomenon is derived (solely) from data. In a recent variation, the theory is derived from data but the coding scheme is seeded with a predefined ontology [Lamp & Minton 2007].

The content account development and validation occurs in three stages: the initial formulation of the content account, refinement and validation of the content account, and analytical application of the content account. During the initial formulation of the account, a single coder (myself) used the ontologically grounded theory method to analyze a series of design documents drawn from a 2009 class project. During refinement and validation a two-coder system was used to elaborate the coding schema using a new data set from a 2010 homework assignment. Independent coding of a subset of the documents and inter-coder tests are to validate the reliability and coverage of the coding

scheme. During analytical application the refined SR.BID schema is used to code and analyze a third set of data from a different 2010 homework assignment. The analytical application uses a more conservative dual-coding methodology, and a twelve week delayed intra-coder test to validate the reliability and coverage of the coding scheme. For easier reference, content account studies use the “C” prefix in their designation, as shown in Table 3-3 which summarizes the content account studies.

Table 3-3. Content account studies and associated study type.

Study	Content Account Studies (C)	Type
C.1	Development of SR.BID	Ontologically grounded

The result of the content account study is SR.BID, a robust content account of problem formulation that can be used to support continued qualitative analysis of problem formulation, as well as to support a process account of analogical problem evolution.

3.4.3 Process Account Development Phase

In this phase, I address the second research problem. The objective is to derive a process account of the problem evolution in which to situate analogical problem evolution (APE). To begin, I first create a more detailed descriptive account of the phenomenon, frame in the new SR.BID content account. Data collected in 2009 is analyzed by coding design documents in terms of the SR.BID content account, and analyzing differences in the SR.BID content models over time. This analysis provides a more detailed account of what is transferred during analogical problem evolution, and when it is transferred. This data is then used as the basis for a conjectured process

account, called PE.BID. The PE.BID account is broken into constituent components: problem representations; designer goals; designer strategies; problem transformations, and analogue memory. Each component is justified separately, with some components receiving a deeper treatment than others, as supporting evidence allows. Problem representations are justified using the SR.BID schema, developed in the previous section. Designer goals are accounted for using an existing theory of complex problem solving. Designer strategies are proposed based on the designer goals and my own experience. Designer transformations are likewise conjectured based on the identified strategies, and are then validated against SR.BID coded problem formulations. Finally, a model of analogue memory will be proposed and grounded in past observations, and is justified later through the implementation of a computational system in Chapter 8. For easier reference, process account studies use the “P” prefix in their designation, as summarized in Table 3-4.

Table 3-4. Process account summaries and associated study types.

Study	Process Account Studies (P)	Type
P.1	APE process description study using SR.BID	Coded design analysis
P.2	Developing an account of problem evolution	
	Sub-study: problem content account	Grounded theory
	Sub-study: designer problem strategies	(work-in-progress)
	Sub-study: transformations	Coded design analysis
	Sub-study: designer problem goals	Adapted from existing theory
	Sub-study: problem-solution memory	Computational study

3.4.4 Tools and Interventions

This phase addresses two additional research problems discovered in the exploratory phase of the research. The first research problem is grounded by the need to provide

students with additional support for specifying their design problem in the context of BID. The second research problem takes a broader perspective on the field of BID, and is driven by the need to provide structured data across massive numbers of biological systems for access by designers. It examines the potential for SR.BID to support distributed acquisition of structured data. I will begin by addressing the first of these research problems.

3.4.4.1 *Research Problem 3*

Problem formulation in BID plays an important role in searching for and evaluating analogical sources. I observe that students struggle with problem formulation in BID, and consequently with analogical evaluation. However, there exists no systematic support for problem formulation specific to the BID context.

To address the problem I examine the utility of SR.BID content account developed in study C.1 to provide support for problem formulation and analogical evaluation in the context of the BID@GATech class. From previous experience we know that such tools must have a low investment threshold for student use, must provide perceived value to students and instructors, and students must be able to use the tools with a degree of accuracy and reliability necessary for the supported task. I provide interventions in the 2011 and 2012 course in the form of the four-box method of problem formulation, and the T-chart method of analogical evaluation. These interventions are tightly integrated with the class design, instruction and assignments. After students use these interventions, they are analyzed in two studies, which draw from different sets of data.

In the first study, students in both 2011 and 2012 are asked to use the four-box method of problem formulation as a homework assignment. Student assignments are

evaluated in terms of the ability of students to complete the assignment as expected, and to use the method accurately to represent their design problem. I show that with less than one class period for training, 97% of students are able to complete the assignment with greater than 85% average accuracy. I also show there is no statically significant source of variance between gender, class, or major. I also show that while the four-box method implies a certain style of arrangement (e.g. four-boxes), students that use bulleted lists or natural narrative formats use the four-box method equally well.

In the second study, as part of a take-home final exam students are asked to reflect over their use of tools and methods taught in class. Although which tools and methods students comment upon are left up to them, it turns out that all students comment on either the four-box method, the T-chart method or SR.BID; many comment on both or all three. The comments are categorized and analyzed to determine the perceived value of the methods, relative to expected benefits.

Couched in the success of the implementations of the four-box and T-chart methods, I address the fourth and final research problem.

3.4.4.2 *Research problem 4*

From observations of systems implemented to support BID, we know there is a tradeoff between representational complexity, cost of representation and potential value/tasks supported. Existing support systems do not yet provide a sufficient return on investment such that system builders are able to get widespread adoption. How might these systems acquire structured data on thousands or tens of thousands of biological source analogues that would be necessary to drive value to the design users?

I use the SR.BID content schema and the four-box method to create a web-based distributed application platform capable of collecting structured data from novice designers, with minimal training. I have shown in the previous studies that students are capable of learning and using the SR.BID-based method of problem specification in a short period of time. In this study, I expose students to the SR.BID web-based application, and encourage students to use it in whatever way they choose. Student use is entirely voluntary, on the students own time and bears no weight on student grades. I track the interactions of students with the system to determine whether or not they are capable of encoding structured data about BID problems and biological systems and how quickly they are able to do so. For easier reference, intervention studies use the “I” prefix in their designation.

Table 3-5. Intervention studies and associated study types.

Study	Technological Intervention Studies (I)	Type
I.1	Accuracy of student use of the four-box method of problem formulation	Intervention
I.2	Qualitative analysis of the four-box method of problem formulation	Intervention
I.3	SR.BID Web Application	Intervention

Table 3-6 provides a summary of research problems addressed in each phase of the dissertation, and the studies and study types associated with those problems. Each phase correlates to one research problem. Within each phase, I will explore numerous research questions and hypothesis and I will employ multiple methods of investigation as appropriate. The research problems, research questions, and hypothesis associated with each study are reviewed in their respective chapters, along with implementation specific details of the research method used.

Table 3-6. Summary of research problems and associated studies.

Exploratory Phase (E)		
E.1	2006, 2007 <i>in situ</i> observation of BID classroom	Observational Study
E.2	2007 Test of biological systems on problem decomposition	Exploratory experiment
E.3	2007 Data analysis of design problem decomposition	Coded design analysis
Content Account Phase (C)		
Research Problem 1: While many theories of design problem representation exist, it is unknown to what extent current content theories of design can support the process of design problem formulation and evolution in BID.		
C.1	Development of SR.BID	Ontologically grounded
Process Account Phase (P)		
Research Problem 2: Current theories of analogy account for analogical evolution of solutions; while current theories of problem evolution account for problem evolution in the context of conjectured solutions. Analogical problem evolution in response to existing analogical solutions remains unaccounted for in design theory.		
P.1	APE process description study using SR.BID	Coded design analysis
P.2	Developing an account of problem evolution	
	Sub-study: problem content account	Grounded theory
	Sub-study: designer problem strategies	(work-in-progress)
	Sub-study: transformations	Coded design analysis
	Sub-study: designer problem goals	Adapted from existing theory
	Sub-study: problem-solution memory	Computational study
Intervention Phase (I)		
Research Problem 3: Problem formulation in BID plays an important role in searching for and evaluating analogical sources. I observe that students struggle with problem formulation in BID, and consequently with analogical evaluation. However, there exists no systematic support for problem formulation specific to the BID context.		
I.1	Accuracy of student use of the four-box method of problem formulation	Intervention
I.2	Qualitative analysis of the four-box method of problem formulation	Intervention
Research Problem 4: From observations of systems implemented to support BID, we know there is a tradeoff between representational complexity, cost of representation and potential value/tasks supported. Existing support systems do not yet provide a sufficient return on investment such that system builders are able to get widespread adoption. How might these systems acquire structured data on thousands or tens of thousands of biological source analogues that would be necessary to drive value to the design users?		
I.3	SR.BID Web Application	Intervention

4 EXPLORATORY STUDIES IN PROBLEM EVOLUTION

In this section I will discuss three exploratory studies conducted in 2006 and 2007 in the context of the biologically inspired design classroom. These studies focused on increasing our understanding high level processes of design through the lens of existing cognitive theories of analogical design. While these studies uncovered many phenomena of research interest, I limit discussion of the findings to only those relevant to problem evolutions.

In the context of biologically inspired design classroom, when designers are formulating their problem, both at the time of inception and/or in later phases of conceptual design, they use existing biological solutions as a means of understanding, framing and articulating their human design problem. In study 1.1, I will provide evidence that solutions influence problem formulation at the time of problem-inception, as in **solution-based design**, where the understanding and abstraction of principles from a biological solution are used as indexes to find design problems to which they may be applied. I will provide additional evidence in study 1.1 that solutions influence problem formulation during conceptual design, which will be seen in the reformulation of problem decomposition during compound analogical design.

In study 1.2, I will demonstrate that when student designers are prompted with biological systems, those systems influence problem formulation at multiple levels of functional abstraction. Finally, in the context of the process of solution-based problem formulation in study 1.3, I will show that designers incorporate structural concepts in equal measure with functional concepts in problem decompositions, even alternating between the two. This implies a process of problem understanding in which student

designers move between consideration of biological systems (e.g. a bird) or aspects of the systems (e.g. a wing), and the function supported by the system (e.g. generate lift). This is reinforced by the process of compound analogy, shown in study 1.1.

This claim is significant in that, despite its frequency, current theories of analogical design, of biologically inspired design and of problem-solution coevolution do not directly account for it. Claim 1, supported by the set of findings from these studies, is critical in that it forms the premise for the two core problems addressed in this dissertation.

4.1 Study E.1: Observational Studies of the BID Class

4.1.1 Study Motivation

In 2006 I entered into an exploratory *in situ* cognitive study of the classroom research environment (Darden & Cook 1994; Dunbar 1995; Kurz-Milcke, Nersessian & Newstetter 2004; Christensen & Schunn 2008). While there were no explicit research questions at the outset, to quote Miles and Huberman (1994) “any research, no matter how unstructured or inductive, comes to fieldwork with *some* orienting ideas.” In this case that frame of orientation includes cognitive theories of analogical design and related frameworks. Thus my goal for these studies was to provide descriptive accounts of the products and processes of biologically inspired design, which appeared to be fundamentally analogical at the outset, through the lens of our theories of analogical design. Although this study was conducted in the context of a classroom setting, I approached the study from a design cognition perspective as opposed to a learning sciences perspective. That is, I was less concerned about the pedagogical approach and the learning outcomes of the course. Although I believe that my research will have

implications on the approach and conduct of the course, I was not directly involved in the decision-making regarding the design of the course. From my perspective the classroom provided a setting where we could observe designers engaged in biologically inspired design.

4.1.2 Methodology of Study

4.1.2.1 Context and Participants E.1

This study takes place in the context of the BID@GATech class, during the fall semesters of 2006 and 2007. The course met twice weekly for 1.5 hours, and each classroom session was attended by the researcher. See sections 3.3.1 and 3.3.2 for descriptions of the context and student designer participants.

4.1.2.2 Materials E.1

This study was observational. There were no experimenter generated materials used for this study.

4.1.2.3 Execution of the Study E.1

Two research scientists participated in the 2006 observational study. As an observer, I attended the classroom sessions, collected course materials, documented lecture content, and observed teacher-student and student-student interactions in the classroom. I had no influence on the course design or pedagogical approach. I also did *in situ* observations of a few of the teams engaged in their design projects. I minimized my intervention, only occasionally asking clarifying questions.

My observations focused on the cognitive practices and products of the designers. I observed and documented the frequently occurring problem-solving and representational activities of designers as part of the design process. Some of these activities were part of

the design process explicitly taught by the instructors. Others emerged during practice. In terms of the design products, I observed and documented the natural evolution of the conceptual design over time. I also attended the final oral presentations of the design teams to the class, and read the design briefs the teams submitted with their projects.

4.1.2.4 *Data Collected E.1*

Observations from both classroom and design observations were noted in a field journal during class. Subsequent reflections and discussions with other participating researchers about observations in the class were also logged in the field journal. Final design reports and presentations were collected and retained in electronic format. Design trajectories such as the project described in section 3.2.4 were synthesized from the breadth of data collected.

4.1.2.5 *Method of Evaluation E.1*

Documents were analyzed through the lens of existing theories of analogical design (Clement 2008; Dunbar 2001; Gentner 1983; Gick & Holyoak 1983; Goel 1997; Hofstadter 1996; Holyoak & Thagard 1995; Keane 1988; Kolodner 1993; Nersessian 2008.). Analysis and synthesis of observations occurred through weekly conversations with an additional research scientist, and were refined and vetted through conversations with a third researcher. Findings were limited to a report of observations. Each finding was supported by multiple student design documents.

4.1.3 Results of Study

4.1.3.1 *Solution-based design process*

Of the biological inspired design projects observed in 2006, five out of nine followed a modified version of traditional problem-driven design process, in which the design

process progresses from an initially described design problem to a design solution. Surprisingly, the other four projects began with a solution-based design problem. In this case, the design process began with a biological system or “solution” (such as impact-resisting abalone shell), and progressed from there to identifying a problem for which the biological system might provide a useful solution.

Thus, while the normative biologically inspired design process pointed to a problem-driven approach, we observed that the biologically inspired design process also progresses from solution to problem, each following a distinct patterns, solution-to-problem or problem-to-solution (Helms et. al. 2008). Some classroom exercises, and many of the case-studies provided to the class, began with a biological solution, extracted a deep principle, and then found problems to which the principle could be applied. In general, the solution-driven biologically inspired design process follows the steps listed below (note again that this pseudo-algorithm only illustrates the high-level pattern of the problem-driven process; in practice, the actual process is not necessarily ordered linearly). Complete documentation of the process of solution-based design can be found in Helms et al 2008.

The identification and description of the solution-based design process demonstrates that very early on in the process of design, knowledge of a biological solution can influence the formulation of a design problem. In the case of solution-based design, the design problems are identified at the outset based on some set of relevant features (such as function) that they have in common with a biological source. Moreover, the fixation of certain aspects of the design solution will cause similar fixation in the design problem. This manifests in a kind of confirmation bias, directing the attention of the designer to

those aspects of the problems that are most in alignment with the already selected biological solution, while causing students to dismiss issues that are in disagreement until they are forced to confront them, usually late in the process (Yen et al 2011).

4.1.3.2 *Compound Analogy*

Another key finding in the 2006 observational study is the description of the process of compound analogy. While some models of analogical design address the issue of cross-domain retrieval and transfer of knowledge (e.g., Bhatta et al. 1994), most existing models of analogical design are single source-based solution generation models. That is, given a target design problem, the process proceeds to retrieve a suitable analogue and modifies or adapts the retrieved design to generate a solution to the target problem. From the cases of BID observed in the 2006 observational study, it is apparent that this form of single-source analogical design is not adequate for generating complex designs. In complex design tasks, multiple sources are often needed to solve different parts of a complex problem. Fully two-thirds of the designs in 2006 involved multiple biological source analogues. This immediately suggests interplay between two related processes, analogy and problem decomposition.

Solving complex problems by decomposition, where designers break large, complex problems into small, less complex, ones is not new. But when we make the decompositions explicit in the context of analogical design, it becomes apparent how the processes of decomposition and analogy influence each other. Their interplay is fully documented in my work on the high-level conceptual framework of *compound analogical design* (Helms, Vattam & Goel, 2008, 2009). Detailed examples of compound analogical design are presented in Appendix C.

This interplay described in the compound analogy framework demonstrates strong interactions between the conceptualization of the design problem and the biological analogue(s) used to frame the solution. In the framework of compound analogy we see that analogical sources of inspiration may not only lead to solutions through direct transfer to a conjectured solution, but also through contribution to the understanding and the framing of the problem itself. The discovery of a biological solution which addresses a problem under a sub-set of conditions, can force the problem to be restructured into conditions representing these sub-sets. A previously unconsidered condition can become a crucial concept in the design problem. In the case of a stealthy underwater robot, for example, the realization that a biological source moved stealthily but only when moving slowly, forced designers to consider different conditions of speed. The problem became partitioned along low speed and high speed conditions.

4.1.3.3 *Common Errors*

In this observational study I enumerated a number of common errors that student design teams make. Among them, three are of particular interest:

Error 1. Vaguely defined problems.

Problems that are nebulously defined, such as “lowering our dependence on oil,” or “protecting a cell phone,” are either too vague to yield to functional descriptions (resulting in no found analogies), or result in too large a search space (resulting in too many poorly matching analogies). Instructor feedback defined better problem definitions as “more efficient allocation of resources to reduce energy consumed in transportation” and “forming a scratch-resistant coating for cell phones.” This observation suggests that

student designers have difficulty formulating design problems at the correct level of abstraction.

Error 2. Poor problem-solution pairing.

Frequently, designers match problems to biological solutions based on vague or superficial similarity, such as matching “making a better dishwashing detergent” with the “cleaning properties of the lotus leaf.” Although the function “cleaning” is similar, the lotus leaf relies on the structural details of the structure to be cleaned, which a detergent cannot manipulate. This observation suggests *that shallow understanding of the problem, the biological source solution, or both, may lead poor analogy choices.*

Error 3. Misapplied analogy.

When making an analogy, superficial or high-level matches are often forced into an incongruent solution space, yielding flawed solutions. For instance, a two-way traffic optimization algorithm derived from ant foraging behavior, applied directly to a throughput traffic optimization problem yielded an erroneous model. Fixation on this erroneous model resulted in three design revision attempts prior to it being discarded. This observation suggests, that even when a good analogy is selected, *identifying, understanding and adapting to critical differences between the problem and the solution is critical in making an effective transfer.*

4.1.4 Summary of 2006 Observational Studies

The observations above support three key findings.

4.1.4.1 *Finding 1: Designers struggle with design problem formulation in BID*

Firstly, I found that in a design context that stresses innovation and creativity, where designers are allowed to determine their own design problems, student designers struggle to formulate their design problem. I observed student design teams formulate and evolve (incrementally reformulate) their design problem, often with radical transformations which require discarding or modifying existing solution design(s). This struggle is ongoing, and often dramatic.

4.1.4.2 *Finding 2: Design problem formulation evolves in response to biological analogies.*

Secondly, I found that design problems evolve in response to an analogy. I refer to this phenomenon as analogical problem evolution (APE). A design problem may give rise to an analogy that instigates an alteration to the design problem formulation. This new design problem formulation may in turn provide new criteria for retrieving and evaluating new analogies, which may in turn alter the design problem formulation, and so on. I observed that some design processes are solution-based designs, that is, the design problem is defined in terms of a problem that can be solved by an already-identified solution. I also observed the phenomenon of compound analogical design in which multiple biological analogues were applied to derive a solution. In compound analogical design it was found that a biological analogy can initiate a decomposition of the problem in a way that the design team had not yet considered. In one observed case, the design team upon learning of a biological analog with both slow- and fast-moving modes of stealthy movement decomposed their problem of stealthy movement into slow and fast modes. While theories of analogical design in BID account for the observed solution

generation aspects of BID, these theories do not fully account for the *problem* evolution aspect of the APE phenomenon. Likewise, existing theories of design problem evolution do not account for the influence of analogous solutions.

4.1.4.3 *Finding 3: Designers have difficulty finding and making “correct” analogies*

Thirdly, I found that the difficulty in defining the design problem translates into certain difficulties in making analogies. This is not unexpected since analogical theories stipulate a “target problem” which forms the basis for many processes of analogy, from retrieval to mapping to transfer to storage¹⁰. In my initial studies I observed that students both had difficulty (a) finding appropriate analogies, and (b) applying the analogy correctly to their design problem, both of which could be the result of a poorly defined problem (although other causes are also plausible). The challenge of an imprecise or dynamic problem-target is not unique to BID; for example in scientific inquiry, problem formulation is likewise dynamic. Nersessian et al (2009) provide a description of the use of analogy in such a context. Moreover that problem definitions change over time in design is well known. Thus while this observation appears intrinsic to BID, it also generalizes to other domains where analogies are found.

4.2 Study E.2: Biological Source Influence on Problem Decomposition

4.2.1 Study Motivation

Findings on compound analogy and solution-based design from 2006 imply that information from biological analogues can be used to decompose design problems into

¹⁰See Gentner (1983, 1989), Falkenhainer, Forbus, & Gentner (1989), Holyoak and Thagard (1989),

smaller problems. For example, a biological system that operates under two distinct modes for stealthy travel depending on its speed, may inform a designer to break down a similar problem into modes that depend on the speed of travel. In this way, biological analogues that use different strategies may lead to alternative problem decompositions. I conducted an exploratory experiment in 2007 to test whether different biological analogues might lead to different problem decompositions.

This pilot experiment attempted to answer the following questions: (1) do different biological sources create alternative conceptualization of design problems? This classroom exercise had both research and pedagogical goals. As a pedagogical device, the exercise served to (1) educate students on biological systems that might be useful to their design project, (2) familiarize students with strategies for decomposing design problems. The pedagogical goals were realized both by participation in the exercise and by a reflective post-exercise discussion conducted after the exercise. The pedagogical goals served as additional incentive for the students to participate fully in the exercise.

4.2.2 Methodology of Study

4.2.2.1 Research question E.1

To what extent do biological systems influence functional decomposition of problems?

4.2.2.2 Hypothesis E.1

The introduction of biological analogues to student designers will yield greater range of concepts in a functional decomposition of a design problem, than a decomposition without biological analogue prompts.

4.2.2.3 *Context and Participants E.1*

The context for this experiment was the 2007 BID@GaTech class. One week prior to the exercise, the students received 45 minutes of classroom instruction in functional decomposition of design problems. Instructors believed that decomposing a problem into constituent sub-functions, would make explicit a number of potential connections to biological sources of inspiration that may otherwise remain hidden. This should in turn increase the number and diversity of biological sources that could be applied to a given design problem. Aside from the pedagogical benefits, this ensured that students were somewhat familiar with the method of functional decomposition presented during the study.

The experiment was time-bound to 45 minutes on the day of the experiment (roughly half of the class period), and it was required that pedagogical as well as experimental goals were met. Due to time and pedagogical constraints, as well as the exploratory nature of the experiment, a facilitated group decomposition was elected in lieu of a more robust study at the individual level.

4.2.2.4 *Materials E.1*

At the time of the experiment, a packet was provided to each member of the class containing:

- A filtration design problem (the same for each packet)
- A shallow functional decomposition for the filtration design problem
- A separating blank page
- A description of one of five biological systems (see Appendix D for details)

The design problem was paraphrased from a design problem used by a design team from the previous years, and read as follows:

“The design challenge is to conceptualize a portable, stand-alone, home-filtration unit that is more efficient than HEPA or ionic systems, especially with respect to the particles in the size range of 0.01 microns to 100 microns. The designed system will address some of the disadvantages associated with the traditional HEPA and ionic filters. With regards to the HEPA filters, the system will eliminate the need for the expensive replacement filters, and will operate at a lower energy output. The system will also resolve the problems of decreasing efficiency and ozone production seen with ionic filters. Lastly, the designed system aims to create a filtration device that is environmentally friendly by using sustainable materials that are also biodegradable or recyclable.”

Figure 4-1 accompanied the text, providing a diagram of relative airborne particle types and sizes.

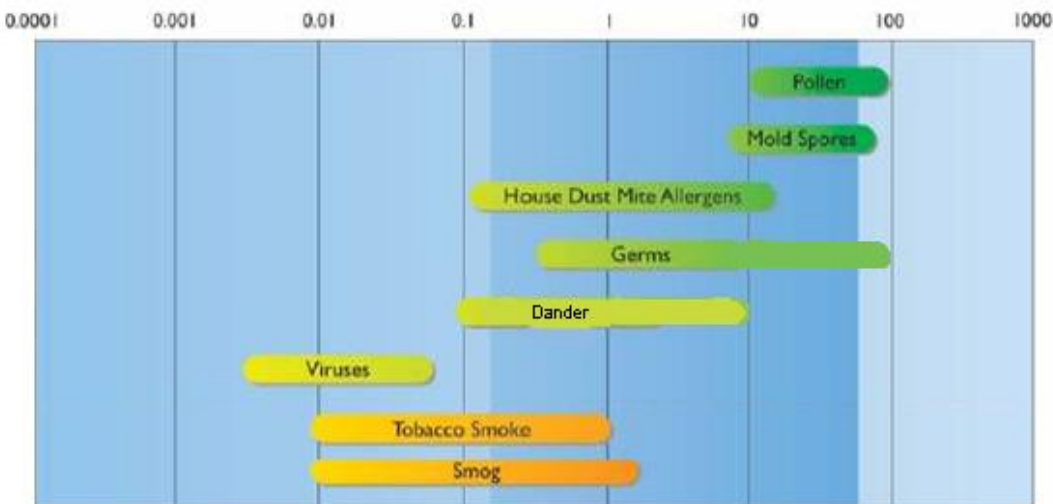


Figure 4-1 Figure depicting particle size range for filtration design problem, included with problem description in study E.1

In addition to the problem, the class was also provided with an initial “shallow” functional decomposition, shown in Figure 4-2. This decomposition was derived from a complete decomposition of this problem which I had generated previous to the

conceptualization of this experiment.

The biological sources included in the packets were taken as a subset of the biological sources described in the final report from the previous year for the same design problem. They included a) mussels, b) baleen whales c) diatoms, d) salps/laravaceans (small jelly-fish like organisms), and e) hemoglobin. Descriptions and images were provided for each source, and were copied from the descriptions provided in the final design report from the previous team.

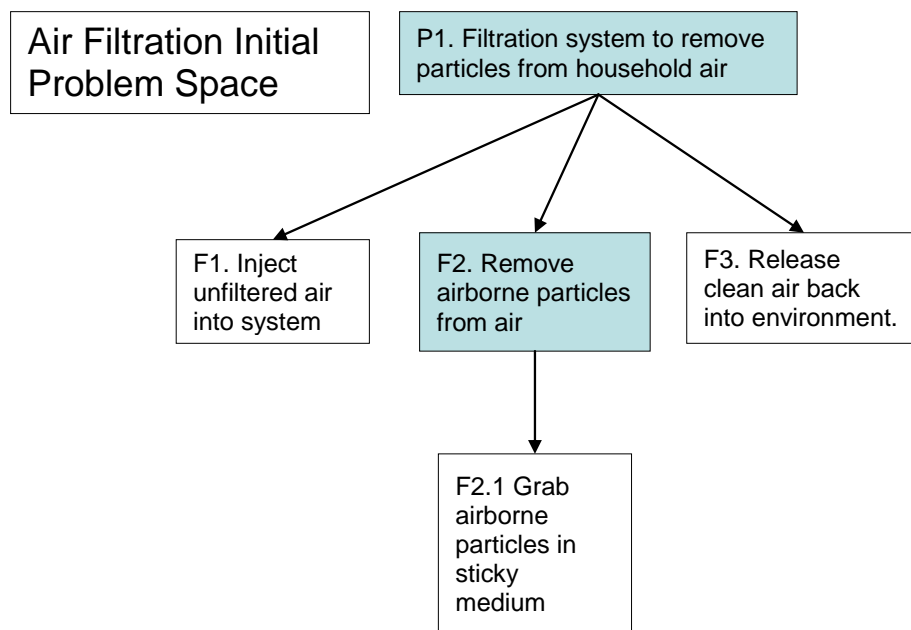


Figure 4-2. Initial “shallow” functional decomposition provided to students for the air filtration problem.

4.2.2.5 Execution of Study E.1

Packets were prearranged, such that each of the five biological systems occurred in

the same rotation and were distributed desk-by-desk throughout the class such that an equal number (+/-1) of biological systems were distributed among the students. After distributing the packets, students were given five minutes to review the design problem and the shallow functional decomposition. Students were then asked by the experimenter to provide a further functional decomposition of the problem. The decomposition was a group activity facilitated by drawing the decomposition on the white board so the entire class could observe. Further, the experimenter filtered the concepts, such that only functions were listed in the decomposition on the board. If a student suggested a concept that was not a function, the experimenter facilitated a discussion to determine if an underlying function was intended. For example, if a student mentioned an issue such as “if the air passes by too fast, the filter might not catch the debris”, the experimenter facilitated a group discussion to arrive at an implied function, such as “control air speed”. The activity was halted when the class replied that they had no further elaborations to make and were satisfied with the decomposition. The state of the board was transcribed.

Students were then instructed to read the material provided on the biological system, and consider the biological system in the context of the design problem. Students were given 10 minutes, after which time, the experimenter asked if they wanted to add anything to the functional decomposition on the white board. The activity was again halted when the class replied that they had no further elaborations to make and were satisfied with the decomposition. The state of the board was again transcribed. The instructors then facilitated a discussion.

4.2.2.6 *Data E.1*

Data consisted of hand-drawn transcription of the state of the board after each

facilitated problem decomposition. Each hand drawn transcription was then translated into an electronic representation, as see in Figure 4-3 and Figure 4-4. Color-coding was

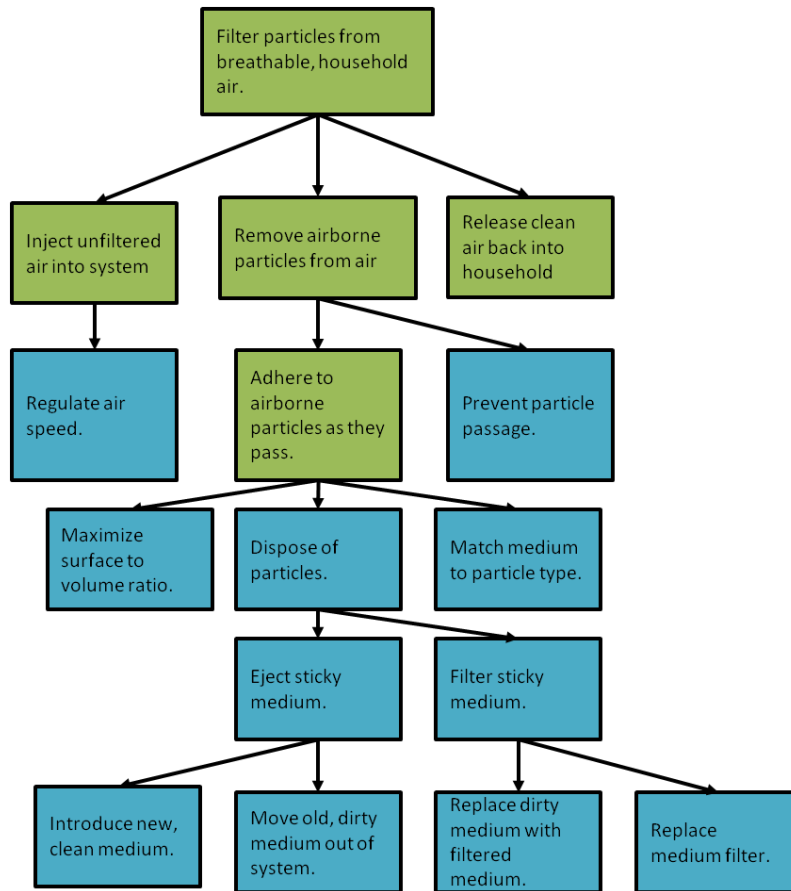


Figure 4-3 Problem decomposition after initial facilitated group decomposition exercise was complete.

added to indicate during which phase each content box was added. All packets that were distributed, were collected at the conclusion of the experiment. Those packets containing annotations were kept. All others were discarded.

4.2.2.7 Evaluation E.1

Before and after functional decompositions were compared, noting that only new functions were added in the second decomposition. For each new function, the author assessed whether the function was also a function performed by one (or more) of the

biological sources provided to the class. The number and depth on the functional decomposition tree of the new functions added, as well as whether or not the new function was related to a biological source was recorded and reported.

4.2.3 Results of Study

This study resulted in a progression of three problem decompositions. The first, initial decomposition was provided to the students, shown in figure 4-2, and carried forward in figure 4-3 and figure 4-4 as green boxes. The second decomposition developed collectively by the class, is shown by the blue boxes in figure 4-3, again carried forward into figure 4-4. In figure 4-3 we see the decomposition deepen by three levels, and the addition of 11 new functions. The majority of these new functions (9 of 11) are focused on the progressive exploration of the seed concept of adhering to the airborne particle as they pass, much of which (6 of 9) is focused on the treatment of the filtration medium as it collects particles.

Table 4-1. Functional concepts added to the second decomposition and the biological organism associated with it.

Concept	Source of Inspiration
Suck in air with bellows	Whale
Use multiple size filters	Diatom
Control water flow	Jellyfish (salps)
Filter water	Baleen whale
Match particle using geometry of particle type	Hemoglobin
Release caught particles	Hemoglobin

Figure 4-4 represents the decomposition after the students were provided with additional biological sources of inspiration: a) mussels, b) baleen whales c) diatoms, d) small jellyfish-like organisms called salps, and e) hemoglobin. The result of the

decomposition after the sources of inspiration shows the addition of eight new concepts added to the functional decomposition, with at least one new concept added at every level of the decomposition beneath the root node. Six of the eight newly added concepts are *directly associated with particular biological sources of inspiration*. Table 4-1 shows the concepts and the associated biological source of inspiration.

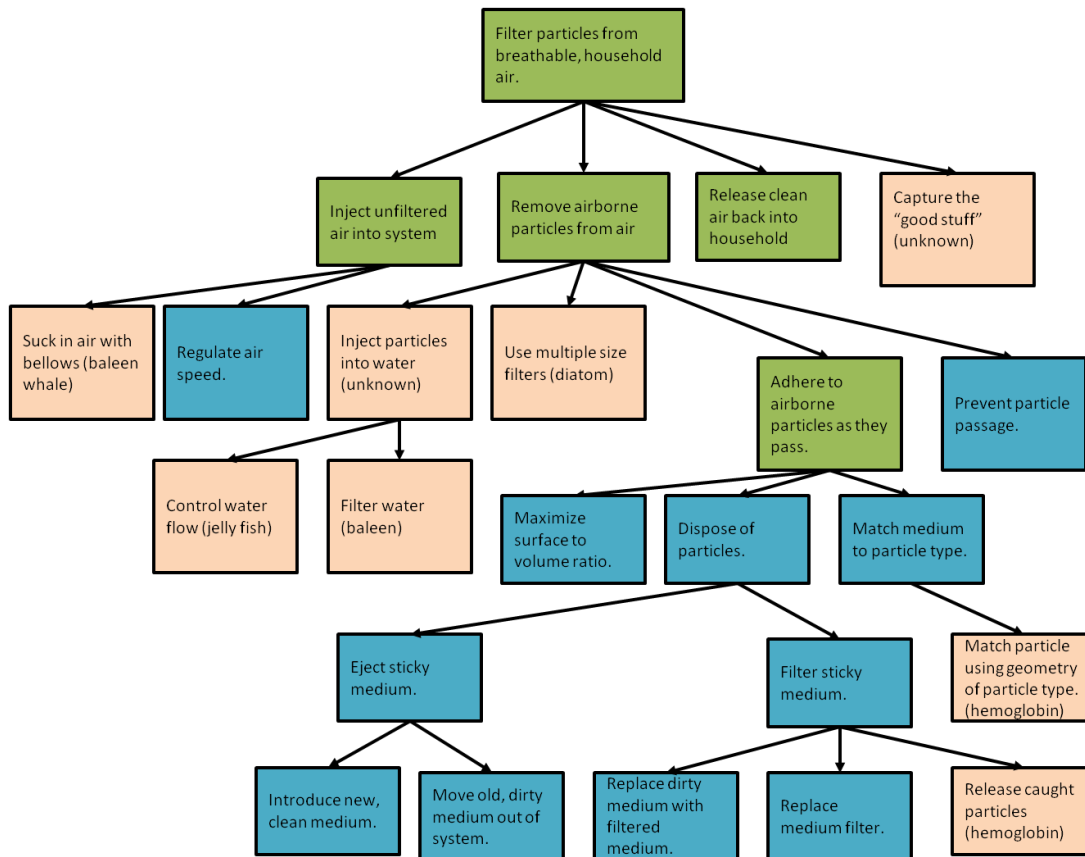


Figure 4-4. The final problem decomposition of a filtration design problem created during an in-class exercise. Green boxes represent the initial (given) decomposition, blue represent the decomposition after a single iteration, pink represent the decomposition after students were provided with biological analogue systems.

The sources of inspiration for the remaining concepts could not be determined. This exploratory experiment demonstrates in principle the connection between the conceptualization of a design problem and a source of biological inspiration; in particular that different biological sources of inspiration lead designers to evolve their problem formulation along those branches of the formulation related to that source. Moreover the addition of new concepts appears to occur at all levels of abstraction in the problem formulation. This implies that concepts transferred from an existing biological system to a problem formulation range from high-level patterns or strategy (“capture the good stuff”), down to the level of implementation (“match particle using geometry of particle type”).

4.3 Study E.3: Qualitative Analysis of Problem-Solution Decompositions

4.3.1 Study Motivation

While the results of Study E.2 looked at the influence of biological analogues on a group problem decomposition, the decomposition was facilitated by the experimenter in such a way as to exclude non-functional concepts from the decomposition. A brief review of homework assignments revealed that when students were asked to provide a functional decomposition of a problem, the decomposition was rarely purely function. The intent of this study is to understand the complete range of concepts that student designer in problem decomposition.

4.3.2 Methodology of Study

4.3.2.1 Research question E.2

To what extent are student problem decompositions purely functional versus a mix of

functional and other conceptual categories?

4.3.2.2 *Hypothesis E.2*

Student problem decompositions will follow a mixed conceptual decomposition strategy.

4.3.2.3 *Context and Participants E.2*

The participants in this study were those student designers that turned in a problem decomposition assignment from the BID@GATech class. In total, 31 assignments were collected. Section 3.3.1 provides an outline of the student population in the context of study.

4.3.2.4 *Materials E.2*

In 2007 students were asked to provide “boxes and arrows” problem decomposition diagrams for a color-related problem that could be solved by a biological solution. The assignment was worded as follows (emphasis added):

*“Everyone must find and bring in a natural object that has an interesting color and research how it achieves that color and why it achieves that color (purpose). Using this knowledge, practice performing the design processes and approaches (SBF, PDs, engineering matrix) so we can address your questions during the Thursday class. Please **submit individual PDs** of your natural object.”*

Six examples of problem decompositions were provided to the students, as part of the materials provided in the assignment (see Appendix C).

4.3.2.5 *Execution of Study E.2*

To aid in the assignment, students were provided with 45 minutes of training in the problem decomposition process for defining a problem. The training included a review of the six examples found in Appendix C. Both the training and the examples provided

demonstrated instances of functional decomposition, in which all concept boxes of the included only functions. The students were asked to use a single biological solution as their source of inspiration to conduct their problem decomposition. Students were provided with one week (subsequent to the training session) to complete their assignment.

4.3.2.6 *Data Collected E.2*

In total 31 students generated and submitted problem decomposition diagrams for a boxes-and-arrows problem decomposition (PD) diagram. Assignments completed after the deadline were not considered for this study. Each assignment was submitted as an electronic file. Assignments consisted of boxes-and-arrows diagrams that were either hand-drawn and scanned, or generated electronically using commonplace software (Powerpoint, Word, etc).

4.3.2.7 *Evaluation E.2*

I analyzed these problem decomposition diagrams in terms of the characterization of the concepts used in the boxes, using SBF as a guide, and adding categories as required. For each box in a diagram, a category was assigned. While the structure of the diagram was not coded, the structure provided context for each concept which was used as an aid for assigning an appropriate category. A tally of the use of categories provided a distribution of category use for defining problems. Table 4-2 shows the category definitions that were used.

Totals for number of concepts, and concept types were then generated. Each problem decomposition assignment was further classified into one of four categories, based on the ratio of concept types present in the problem decomposition. These categories emerged as

a result of the initial coding, and were not determined beforehand.

4.3.3 Results of Study

Table 4-2 shows the weighted average breakdown of category use. Table 4-3 shows the number of assignments coded into one the four emergent categories, as well as the average number of concepts (boxes) in each.

Table 4-2. Conceptual categories, definitions and the percentage of their occurrence in student functional decomposition assignments, measured over all occurrences.

Category	Definition	Percentage Occurrence
Function	A <i>verb-noun</i> phrase A <i>verb-self</i> phrase (self implied) A <i>biological</i> function (e.g. photosynthesis)	40.57%
Function (refinement)	One or more <i>means</i> of accomplishing the function (e.g. pollination by insects, by air, by hand); One or more <i>prepositional extensions</i> of the function (e.g. movement on water, on air, on land)	5.42%
Structure	A <i>property</i> , <i>component</i> , or <i>material</i> composition of a solution (e.g. the color red, a flower petal, and protein respectively)	26.89%
External Factors	The <i>environment</i> (e.g. in the forest) or a <i>condition</i> of the environment (e.g. partially shaded) external to the system.	5.19%
Solution	<i>Solution</i> is used to perform function, <i>Solution</i> performs function itself <i>Solution</i> described a method for performing function	18.16%
Behavior (causal)	A simple causal phrase (A causes B) A complex causal description	3.77%

While the instruction and examples, based on existing theories of functional decompositions (Chandrasakaran, 1990, Pahl & Bietz, 2003; Stone & Wood, 2009) focused on strictly functional decomposition, the data presents an alternative viewpoint. It appears that, at least when considering problem related to a biological system, from the data analyzed very few (6.5%) of students provided a purely functional decomposition. Instead student designers consider the solution and the structural details of the solution in

conjunction with the functions performed by those solutions, and decompose the design problem in a mixed fashion. The majority (51.6%) of decompositions are mixed decomposition, consisting of the complete range of concepts identified. All of these mixed decompositions include multiple (more than one) references to a solution or solution structure. The next most significant class of decompositions (35.5%), the function-solution class, again emphasize that for problem formulation inspired by a biologically solution, there exists an inter-relationship between the solutions and functions. Considering all of the problem decompositions, 87.1% included solutions and solution structures. When considering the total of all concepts in aggregate, despite instructions and examples of function-only decomposition, 45.05% of the concepts related to structure/solution while roughly the same about 45.99% were functions.

Table 4-3. Decomposition types, definition of type and number of assignments classified as each type.

Decomposition Type	Definition of Type	Number	Average # of Concepts
Purely functional (#15, #22)	The purely functional decomposition aligns with the examples provided in class. We consider any decomposition that uses only functions, or only function plus the source solution as the root node, as a purely functional decomposition.	2 (6.5%)	10.5
Function-solution (#11, #12)	The function-solution decomposition consists of a combination of solution or structure concepts and functions. This class of decompositions often demonstrates a function-solution-function alternating breakdown.	12 (35.5%)	13.3
Solution-focused (#6, #31)	The solution-focused decomposition had either zero or one function, and only solution concepts (either solutions or structures).	2 (6.5%)	6
Mixed (#24)	Mixed decompositions included externalities and/or behaviors in addition to having both functions and solution/structure concepts	15 (51.6%)	15.4

There is a strong likelihood that the solution and structural emphasis observed is

influenced by the student designer's overall design experience. It is reported that novice designers focus on structure (Gellert, 1962; Hmelo, Holton & Kolodner, 2000; Mintzes et al, 1991; Wood-Robinson, 1995), while experts focus on deeper function and behavior concepts (Feltovitch et al, 1992). This should not be construed as counter to the point, but rather supportive. The evidence here agrees with those findings, demonstrating that for student designers, there exists strong interplay between the structural details of a given biological solution and the formulation of the related problem.

4.4 Summary of Exploratory Studies in Problem Evolution

The above studies expose a close interaction between student designer consideration of biological solutions, and the concepts they use when describing their design problem. In my observational studies, students are observed to struggle with design problem formulation. Furthermore, when formulation does occur, it is through complex interactions with biological solutions. Observations bear this out from different perspectives.

The process of solution-based design, occurring naturally in roughly half of the observed cases of BID, depends on an initial seed biological source from which a principle may be extracted and which in turn prompts problem inception. Compound analogy, occurring equally as frequently, entailed the use of multiple analogues in the development of a solution to a system. A compound analogy is often the result of a partitioning of a design problem into independent sub-problems each of which can be addressed by a different biological source. The cause for this partitioning is often a biological source itself, as in the stealthy, but low-speed copepod in the example.

The experiment in problem decomposition demonstrated that when student designers

are prompted with biological analogues, they are capable of redefining a design problem at almost any level of abstraction. Finally, I show that in solution-based problem decompositions solution-dependent concepts such as the parts or materials of a biological system, serve as fundamental conceptual components of student problem formulation, occurring equally as frequently as functional concepts. Taken in combination, this evidence demonstrates that when student designers formulate problems in the biologically inspired design classroom context, beginning with problem inception and continuing throughout conceptual design, biological analogues influence problem conceptualization. It is from these studies that my first claim, and two research problems arise.

4.5 Claim 1

Analogical problem evolution (APE), the incremental alteration of a design problem in response to a biological system, is a fundamental characteristic of the process of biologically inspired design.

4.6 Research Problem 1

While many theories of design problem representation exist, it is unknown to what extent current content theories of design can support the process of design problem formulation and evolution in BID.

The first problem arises as a consequence of the need to systematically examine design problems. Existing content accounts of *design problems* in design theory are generally either (a) prescriptive, as in requirements gathering protocols, or (b) extremely abstraction (e.g. functions and constraints). For a study of problem evolution, we require a lens by which the problem may be systematically examined. We require a rich content

account of design problems that accounts for the relationship between biological analogies and design problems, for the breadth of biological analogies and design problems observed in the classroom context. An account and method capable of systematically describing the changes occurring in a design problem in such a way that one can correlate such changes with the biological analogy. The developed system, a content account of design problems and problem-solution relationships, must also provide sufficient generalization to account for the variety of biological systems and design problem observed. Thus the first problem addressed in this dissertation is as follows.

4.7 Research Problem 2

Current theories of analogy account for analogical evolution of solutions; while current theories of problem evolution account for problem evolution in the context of conjectured solutions. Analogical problem evolution although evident in practice, remains unaccounted for in design theory.

Current theories of analogical design and problem-solution coevolution exist in the context of a conjectured design solution, which are different from existing design solutions, and even more different from design solutions from distant domains. Theories of analogical design, which do consider such distant domain analogies, do not however address the issue of problem evolution as a first class phenomenon. While theories of problem-solution coevolution provide an explanation for problem evolution in the context of a conjectured solution, no current theory of problem-solution coevolution provides an account for how a biological analogy may exert direct influence on the evolution of a design problem in the absence of such a conjectured solution.

This concludes the exploratory phase of this research. In this phase three studies led to two main research questions. The first of these questions, regarding a content account of problem formulation in the context of BID, will be address in chapter 5. The second question regarding a process account of APE will be addressed in chapter 6. In chapter 7, I will proceed with the investigation of tool support in the context of biologically, using the results of chapters 5 and 6 as guides for the implementation of tools.

5 SR.BID: A CONTENT ACCOUNT OF DESIGN PROBLEMS

This section addresses the first research problem posed in this dissertation. The designation of all studies and associated research questions and hypotheses in this section will be preceded with the character “C” (for content).

Problem 1: While many theories of design problem representation exist, it is unknown to what extent current content theories of design can support the process of design problem formulation and evolution in BID.

To address this problem, I will first explore the existing content accounts of problem formulation in design. From this exploration, I settle on an underlying content account which I then use as the basis for building out my own content account of problem formulation, which I call Structured Representations for Biologically Inspired Design (SR.BID). I will end this section with the *claim* that my content account, SR.BID, provides a reliable, comprehensive account of problem formulation within the context of study.

5.1 Existing content accounts of problem formulation in design.

5.1.1 Motivation

Many diverse theories of design exist, and most of them discuss design problems to some extent. As with any good design, each theory is designed to fulfill certain functions for a particular set of users in a particular research environment. As a result, while each may contain in part some conceptualization of a problem formulation, the goals,

motivations, contents, processes and levels of abstraction of each theory will vary according to the needs of the designer and community. I am looking to develop a content account for problem formulation in design that meets the set of criteria unique to building theories of analogical problem evolution in biologically inspired design. I use a set of criteria against which to judge existing theories against my needs. The goal of this study is to identify the theory that best meets the needs, in order to serve as a seed ontology for further development.

The degree to which a design theory may be considered to support a cognitive theory of design problem evolution in BID may be inferred based on six criteria: (a) the taxonomy of problem concepts, (b) the taxonomy of problem concept relationships, (c) support for the biological domain, (d) support for processes of analogy, (e) support for processes of problem-evolution, and (f) support for cognitive models.

5.1.2 Evaluation of Theories

This study is based on a literature review. The data used in this study are the research papers gathered in the literature review, documented in chapter 2. These research papers were gathered in the course of a standard literature review process, from scientific and peer reviewed books, journals, and conference proceedings.

Only design theories that make claims about a content account of problem formulation are considered. Each existing design theory is first categorized into one of four main types of design theory. These theories of design emerge from the study of the theories, and are not based on an a priori categorization schema. Table 5-1 shows theory author, and the category to which the theory was assigned. The four categories are defined as follows:

Normative: Normative theories of design are based on observation of best practices in a particular design environment. These best practices are summarized and articulated as a process or set of tools that should be followed in order to achieve good design results. The theories are usually, but not always, grounded in the pragmatic concerns of day-to-day design.

Normative-functional: Normative-functional theories are centered on the idea that specifications that are function-oriented lead to good design. Function orientation may be seen as a solution-neutral abstraction, which provides access to a wider variety of solution concepts than might other conceptualizations. Normative-functional theories suggest a set of tools or processes for specifying a problem primarily in terms of functions, with additional secondary features. These theories often advocate a particular ontology of functions as optimal or comprehensive for representing a problem or solution.

Abstract, computational: Abstract computational theories are those which seek to understand problem formulation in computational terms. Often these theories draw on earlier work in artificial intelligence and mathematics, and describe problems and design processes in the computational vocabulary of search and search spaces.

Solution-oriented: Solution-oriented theories refer to those theories that seek to develop computational tools to support solution design. These theories usually focus on rich representations to support reasoning over solutions and solution development, while proposing static problem definitions that are impoverished relative to the conceptual depth offered for solution reasoning.

Table 5-1. Design problem formulation theory categories, and the authors associated with them.

Category	Citation
Normative	Wirth (1971), Dahl, Dykstra, and Hoare (1972), Roozenberg & Eekels (1995)
Normative-functional	Altshuller (1984), Struges et al, (1996), Kirshman and Fidel (1998), Hundal (1990), Stone & Wood (200?), Pahl& Bietz (2003)
Abstract-computational	Simon & Newell (1972), Simon (1973), Gero (1993)
Solution-oriented	Goel and Chandrasakaran, 1989; Goel, 1992; Bhatta and Goel, 1994; Gero and Kannengiesser, 2004; Sarkar ad Chakrabarti, 2008

I evaluate each categories of theories with respect to my six criteria. The evaluation of each theory category is based on evaluating the capability of *any* theory to fulfill the requirements of the variable. Each category is ranked on a three point scale: full support, partial support or none (does not support). This evaluation establishes the general extent to which each theory category provides an underlying cognitive account for problem evolution in BID. The following table highlights the results.

5.1.3 Results of the Evaluation

The results of the study are summarized in table 5-2, which shows that no single theory or category of theories fully supports a comprehensive content account of BID. While many different problem categorization schemas exist as prescriptive methods for *soliciting customer requirements*, the goals (design requirements that lead to design stakeholder satisfaction) and the process (solicitation and evaluation of requirements) for using these systems are different. This table shows that no theoretical system exists “out-

of-the-box” that meets the requirements for representing both biological systems and engineering problems, with the reliability required in this context. It also shows that comparatively, solution-oriented design theories provide a higher level of support than others. Of these, SBF and SAPHhIRE models appear to be most promising, both in terms of the breadth of supported concepts, and because they have both been applied in the domain of biologically inspired design.

Table 5-2. Amount of support for a cognitive theory of analogical problem evolution, measured in terms of full support, partial support or no support, for each of six variables provided by each category of problem formulations

Variable	Normative	Normative-Functional	Abstract, computations	Solution-oriented
Categories	Full	Partial	None	Partial
Relationships	Partial	Partial	None	Partial
Biological Domain	None	Partial*	None	Full
Analogical Focus	None	Partial*	None	Full
Problem Focus	Full	Full	Full	None
Cognitive Focus	None	Partial	Full	Full

* recently developed theories and applications support some aspects of BID

5.2 The Use of SBF in the Biologically Inspired Design Classroom

In addition to support from the literature evaluation, the instructors of the class also adopt the SBF vocabulary for structuring assignments. The following provides a detailed description of the use of SBF in the biologically inspired design classroom.

In 2007 the instructors introduced SBF analysis as a framework for organizing found object exercises. Students were asked as part of the Found Object homework assignments

and in their discussions to (a) focus on a single *function* of the organism in question, (b) identify the *structures* relevant to accomplishing that function, and (c) provide a *behavioral* explanation for how those structures give rise to the function. Instructors facilitated these discussions as necessary to guide students. (In the SBF vocabulary, behavior is synonymous with causal mechanistic explanation.)

As expected, students discussed structure at length, although they were unable to limit themselves to the discussion of a single function. SBF is a hierarchical representation and systems are naturally functionally hierarchical. As a result it was difficult for students to maintain a single level of functional abstraction during their discussions. Often students travelled “up” the functional hierarchy attempting to explain *why* the organism performed the function in question such as reproduction, survival, escape from predators, etc. The result was discussions about many high-level functions that lacked in detail. Less frequently, students travelled “down” the functional hierarchy, explaining a small portion of *how* the organism performed a function. These discussions usually resulted in very detailed, technical low-level discussions that only a few students could follow. One must continually emphasize to the students that, while the number of levels in a decomposition is very large, functions expressed at much lower or higher levels than the original problem may not always be useful for the purpose at hand, because they introduce constraints (lower levels) or goals (higher levels) not present in the initial problem definition. In addition to traversing levels of abstraction, students frequently confused the different senses of the word “behavior.” Students often associate behavior with higher-level actions at the organism level e.g. mating behavior, territory marking behavior, seeking shelter from the heat, etc. rather than addressing the causal mechanisms, as this

word is used in the cognitive sciences (Gero & Kannengeisser 2004; Goel, Rugaber & Vattam 2009).

To simplify the vocabulary, in 2008, instructors changed the SBF vocabulary to a What-Why-How vocabulary, mapping “What” to “Structure”, “Why” to “Function”, and “How” to “Behavior.” This was an attempt to both remove the ambiguous interpretation of “behavior” and to formalize the levels of functional abstraction. Functional abstraction was considered in terms of “why” moving up the hierarchy (more abstract, super-functions), and “how” moving down (more detailed, sub-functions). Again, students were asked to describe all biological systems in these terms, both conversationally and in formal homework assignments and design reports.

As a rule-of-thumb, instructors have found that restricting the analysis to one to three hierarchical levels above or below the ‘what’ function” is useful to focus the student’s attention on the right structures, functions and mechanisms. Levels above this cut off often take the students to the ultimate evolutionary objective of a given biological “solution”, which may not match the engineering problem for which a given function may be useful. For example, the ultimate evolutionary objective: to survive, is so universal that it gives no additional guidance in the search for connections between biology and engineering. Going too many levels down may introduce constraints specific to the particular way the biological function is achieved, and which may not be relevant if the goal is to abstract the function rather than copy precisely the mechanism.

Thus we see that in both 2007 and 2008, the SBF representation or the corresponding What-Why-How representation provides constructs for ordering student assignments and discussions. The use of SBF representations in one form or another has continued in the

biologically inspired design classroom since 2007, suggesting a continued utility to the instructors. SBF representation provides a reasonable starting point for the development of an ontology capable of representing design problems and the relationship to solutions in the context of the biologically inspired design classroom.

These observations, in conjunction with the literature review, support the development of a model of problem formulation for BID using the underlying SBF ontology. While SBF should partially support problem formulation for BID, some adaptations may be required to support the phenomena of interest and the observed data. In the next section, I will describe the use of the SBF ontology as a basis for the development of SR.BID using the ontologically grounded theory method over two sets of data. I will validate this account using a third set of data, and conclude with the *claim* that SR.BID provides a valid and comprehensive content account of problem formulation in biologically inspired design.

5.3 Study C.1: SR.BID Schema Development

In this section I develop the complete SR.BID account of problem formulation in biologically inspired design. There are three measures of success for the account: (1) the ability of the account to reliably attain the same encoding for a given analogue/problem, (2) the ability of the system to attain results across the range of biological analogues and mechanical engineer design problem observed in class, and (3) the ability of the account to enable analysis of the correlations between design problems and biological analogues. The account, however, cannot be held apart from the system of encoding. Thus the measures of success apply not to the content account alone, but to the underlying system of encoding. In this way I distinguish between SR.BID as a content account of design,

and the SR.BID Encoding Scheme as a systematic method of encoding data using the SR.BID content account.

5.3.1.1 *Research Question C.1*

What adaptations to solution-oriented theories are necessary to support a content account of design problem formulation and evolution in BID?

5.3.1.2 *Hypothesis C.1*

SBF provides a partial content account of analogical design that may be used as a seed ontology to discover the underlying account of problem formulation and evolution in BID.

5.3.1.3 *Context and Participants C.1*

Data for this study was collected in the context of the BID@GATech classroom in 2009 and 2010. Participants vary by year and assignment. See sections 3.2 and 3.3 for a complete description of the 2009 and 2010 classroom contexts.

5.3.1.4 *Materials C.1*

The details for the assignment materials used as prompts for students are documented in section 3.3.3 and 3.3.4.

5.3.1.5 *Execution of Study C.1*

I use a modified form of grounded theory (Glaser & Strauss 1967; Strauss & Corbin 1990), called ontologically grounded theory (Lamp & Minton 2007), to show that SBF as a seed ontology may be applied to problem formulation data to form a comprehensive account of the content account. The account is enriched by adding, modifying or deleting concepts and relationships as required, through several iterations.

In the Grounded Theory methodology, a theory about any phenomenon is derived

(solely) from data. In the ontologically grounded variation, the theory is derived from data but the coding scheme is seeded with a predefined ontology. Using a seed ontology previously established in the domain of study, provides a source of external validation for the foundation of the schema, at the risk of generating a schema bias. The bias of the seed schema will be somewhat, though not altogether, mitigated through a number of repetitions of multi-coder analysis using a range of classroom examples. In part, maintaining the biases generated by using the seed ontology helps ensure the schema is at least partially compatible with existing theories of biologically inspired design that are grounded in the ontology. Thus the selection of the methodology and underlying seed ontology strikes a balance between external validity and internal expediency and utility.

I will use SBF to seed the coding scheme and then derive SR.BID from data about biologically inspired design. The coding will take place in three phases. In phase one, I will use one set of data establish the correspondences between the SBF seed ontology and the data, and develop high level categories for data that does not align with the seed ontology. In phase two, I will use a second set of data to establish a rich account of categories and sub-categories, and relationship among them, validating the reliability of the SR.BID schema using standard inter-coder reliability techniques. In the third phase, I will use yet a third set of data to conduct and analysis of problem statements.

5.3.1.6 *Data C.1*

For this experiment I use three data sets for establishing and validating the schema, which I will call Structured Representations for Biologically Inspired Design, or SR.BID. All three data sets were gathered in the context of the Georgia Tech class on biologically inspired design class. As described in sections 3.3.3 and 3.3.4 of the dissertation the first

set of data is of the *student design project* (3.3.4) type of data, while the second and third data sets are strictly *problem definition assignments* (3.3.3).

The first set of data consisted of the project submissions of one design team in 2009 that focused on solar energy capture for use in homes. The project was selected as a typical example of biologically inspired design, in that it received neither an exceptionally low nor high grade, it experienced critical feedback from instructors on par with the feedback received by other teams, and the team demonstrated steady and progressive deepening of the design in a consistent direction (from abstract to concrete over a cohesive problem domain. The data consisted of 4 individual problem description assignments, a team mid-term presentation, and the team final presentation. For individual problem description assignments each designer generated a 1-2 page text description of their interpretation of the design problem the team was working on. In the case of the mid-term and final presentations, the complete text descriptions in the slides used during the presentation were used as data. In both presentations, teams were required to describe their design problem. Only the text related to the definition of the problem was used. I refer to this as the **2009 data set**.

The second set of data we used consisted of an individual assignment to students in 2010. This assignment asked students to provide a short 1-2 page design problem description suitable for the biologically inspired design context. A total of 38 assignments were collected, one was eliminated as it belonged to a member of our lab who was taking the class at the time. We consider him too well versed in our theories to be considered an unbiased source. These assignments were collected in the third week of class. I refer to this as the **Week 3 2010 data set**.

The third set of data we used consisted of an individual assignment to students in 2010, collected during the eighth week of class. This assignment consisted of problem descriptions between one quarter of a page and one full page in length. A total of 32 assignments were collected, including the assignment from the member of our lab, which was again eliminated. I refer to this as the **Week 8 2010 data set**.

5.3.1.7 *Brief Review of SBF*

Because the method selected is dependent on the SBF seed ontology, here I provide a very brief description. The SBF ontology consists of three nested high-level models, *structure*, *behavior*, and *function* models (Goel, Rugaber & Vattam 2009). The structure model consists of a set of elements, which may be classified as *elements* such as *substances* or *components*, and *connections* among them. *Elements* may have associated *properties* and *values*, while *connections* express the relationship type (e.g. hinged) between *elements*.

The *behavior* model consists of *states* and *transitions* between the *states*. *States* consist of a set of *elements*, and a set of *property - value* for the *element*. Each *transition* is annotated by *causal explanations* for the *transition*. Since one kind of *causal explanation* pertains to a *function* of a *component*, *behaviors* act as indices to *functions* of *components*.

The *function* model consists of a *given* or *prerequisite state*, and one or more *makes* or *resultant states*. It also specifies one or more external *stimuli*. In addition, it specifies the *behavior* that accomplishes the *function*. Thus, functions act as indices to behaviors.

Table 5-3 represents the initial SBF ontology we begin with for coding our design documents. The SBF coding scheme also suggests the grain size at which the design

documents should be analyzed. In this case we use a coding structure comprised of up to several words at a time. For instance a function typically presents as a verb-noun pair, such as “clean surface” or “generate lift.” A component may present as a word such as “leg”, “muscle”, or “wing,” whereas a property-value pair may present as a short phrase such as “positioned at 32 degrees.”

Table 5-3. Conceptual “seeds” of the SBF ontology

STRUCTURE	BEHAVIOR	FUNCTION
ELEMENT	STATE	PREREQUISITE STATE
--Component	--Element	RESULTANT STATE
--Substance	--Property & value	STIMULI
PROPERTY	TRANSITION	
--Value	--Causal explanation	
CONNECTION		

5.3.2 Phase 1: Initial Coding of SR.BID

During initial coding, my goal was to align the SBF seed concepts with the data and add new conceptual categories as they emerged from the data. In this early phase, I coded the data myself to map the problem description text data in the 2009 data set to the conceptual units found in the SBF ontology. Phrases deemed unrelated to the design were omitted from the analysis. After SBF based codes were assigned, patterns were identified in the remaining phrases (such as frequent discussion of the deficiencies or benefits of existing designs), and categories developed (for example, deficiencies and benefits). As an example, the following is an excerpt from a problem description:

“I think this is a big gap between the static and fragile solar panels that we have so far engineered. So far, most solar panels are set up on a grid basis acting together especially when moving to the sun rather than as individual. Continuing off that tangent I think it would be interesting to have an individual solar panel that can stand alone and still

function. The snail shell structure is stand alone and has the ability to passively dissipate heat by using the heat gradient so that it is cooler within the shell than the outside air and ground this would be helpful for allowing the interior of a structure with solar panels to remain cool.”

In all, 2405 words were coded, chunked into 636 concepts. Of these, 66 (10.4%) were deemed unrelated to the design project such as the phrase “continuing off that tangent” in the above excerpt. I analyzed the remaining 570 concepts, which exposed several new categories for inclusion in SR.BID.

Of the concepts that matched categories directly in the SBF ontology, by far the largest number of concepts represented *functions*. In the excerpt above “act together”, “track sun”, “dissipate heat”, and “remain cool” are all straightforward functions. This agrees with the common wisdom that functions play a key role in defining a design problem. In all 155 (27%) concepts were classified as functions.

A large percentage of concepts were high-level references to *solutions*, either man-made (e.g. “solar panels”, “heat towers”) or biological (e.g. “desert snails”, “diatoms”). In the above example, designers are looking at “solar panels...set up on a grid basis”. They also include the existing biological “snail shell structure” in their description of the design problem. These high-level references to existing *solutions* in the context of the problem description data accounted for 145 (25%) of the total concepts encoded.

Given the number of concepts related to existing solutions, we expected to see a discussion of concepts analogous to the SBF structural model of these solutions. However, there were only 26 (4.6%) structure model concepts. For example “the structure of the diatom shell: pores, spicules, and channels” was coded as four distinct structural *components*: diatom shell, pores, spicules and channels. Moreover, with respect to the *components* in this case study no properties or values were discussed nor were

explicit structural connections among components of the kind one would find in an SBF model. There were, as in the case of the diatom shell example, part-of relationships i.e. pores, spicules, and channels are all part of the structure of the diatom shell.

Behavior descriptions were likewise uncommon. Most of these behaviors were in the form of equations: either chemical reduction equations, for example “ $2\text{H}_2\text{O} \rightarrow \text{O}_2 + 4(\text{H}^+) + 4(\text{e}^-)$ ”, or mathematical expressions of physical phenomena such as “efficiency = $R(\text{T}_1 - \text{T}_2)$ ”. Two others were explanations of complex phenomena for example, “dirt particles are picked up by water droplets due to a complex micro- and nano-scopic architecture of the surface”. In this case, structures (“dirt particles”, “water droplets”, “surface nano-scale architecture”) and functions (“picked up”) embedded within the behavior and were coded separately, in addition to the coding the entire phrase as a behavior. In total there were 16 (2.8%) such examples of behaviors.

As might be expected in a problem statement, constraints on components also played a role, although less than we expected. In total, only 26 (4.6%) concepts were related to the device itself, 8 of which were references to cost, 5 of which were related to the use of sustainable or biodegradable materials. The remaining constraints concerned material, weight, size, flexibility, assembly, and nano-scale manufacturability.

Related to existing solutions, I observed judgments of these solutions, both positive and negative. Negative judgments of existing solutions included “lack of availability”, “limited useful lifetime”, “unsustainable” and “taking up too much space,” while positive judgments included “free”, “reusable”, and “sustainable” to name a few. I added these judgment concepts to the SR.BID scheme as *deficiencies* for negative judgments, and *benefits* for positive. A total of 53 (9%) concepts were considered *deficiencies* or

benefits.

I also observed concepts that directly modify *functions*. In the example above, the function “dissipate heat” is modified with “passively” which imposes an additional criterion on the function i.e. not only does the system dissipate heat, but it does so passively. We denote such modifiers as *performance criteria*, and add this category to the SR.BID scheme. A total of 62 (11%) concepts were considered *performance criteria*.

The functional model in SBF also includes the concept of external stimuli, which is conceptualized of as an input to the system from outside of the system boundary. Such external stimuli, however, were not seen per se. What we did detect in abundance however was the concept of an *operational environment*¹¹. In our example above, “the outside air”, “ground”, and “the sun” all represent aspects of the operational environment in which an existing system or the design problem itself exists. In total, 84 (14.7%) of these concepts were detected ranging from users (“food manufacturers”, “household”), to conditions (“cloudy”, “high pressure”, “windy”, “rainy”), to locations (“interior of structure”, “desert areas”), to reactive living entities (“algae”, “fungi”), to temporality (“during the daytime”, “in the future”). In only four instances of the 84 are numerical specifications provided (e.g. “at a depth of 600m”). While not expressed in terms of input to the system directly, these operational environment concepts seem to be standing in as indices into complex environmental models, which in turn can be simulated by the interpreter to generate a variety of potential inputs into the system.

While we did not see *external stimuli* described in the SBF ontology, what we did detect in abundance was the concept of an *operational environment*. In our example

¹¹ I note that ESBF, an extension of the core SBF ontology, does elaborate on operational environment. However, neither the ESBF ontology nor the other extensions (DSSBF, PSSBF, etc.) were used as part of the seed ontology.

above, “the outside air”, “ground”, and “the sun” all represent aspects of the operational

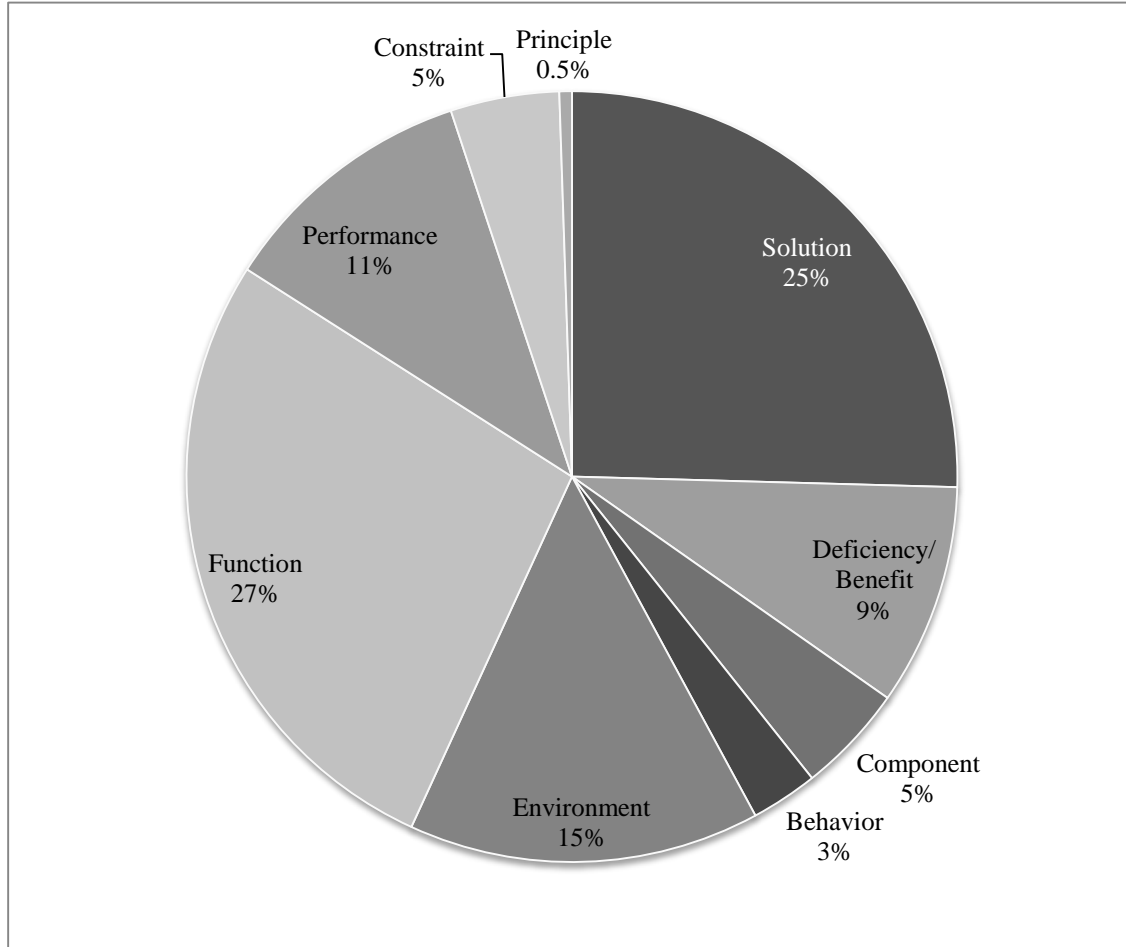


Figure 5-1. Percentage breakdown of 570 concepts by concept type

environment in which an existing system or the design problem itself exists. We include the category *operational environment* in the SR.BID scheme. A total of 84 (15%) concepts were considered *operational environment*

The remaining three (0.5%) concepts were principles, for example “by oxidation”, of the kind defined in SBF as a by-principle transition. Figure 5-9 shows the relative breakdown of the 570 relevant concepts by percentage. That component (5%), behavior (3%), benefits and deficiencies (9%), and solution references (25%) are all grounded in

existing or conjectured solutions, provide additional evidence that solutions play a role in defining a design problem, at least in this context. In this instance 42% of the concepts used in the definition of a problem relate directly to a design solution. And while 27% of the concepts are functions, performance concepts (11%) provide further specification to these functions, suggesting that functions also dominate thinking in problem formulation. Of the remaining 20%, considerations of operational environment take up 15%, and only 5% is dedicated to constraints on the design itself.

When coding at this level some phrases were ambiguous, and required interpretation to categorize. For example the phrase “there is potential clean energy running around” is a metaphor describing a situation in which there exists energy which itself generates no adverse conditions for the environment and which is potentially available at no cost. The phrase is certainly not about the literal interpretation of the *function* of “energy running around,” and is not about “potential energy” as differentiated from “kinetic energy”, but it could be reasonably interpreted in yet new ways. It might be interpreted as a kind of *solution* “clean energy,” or perhaps as a “*clean*” performance criteria regarding an implied *function*, the generation of energy. In this case, my interpretation was as an *operational environment*, a condition in which a solution to be implemented has access to clean energy. A total of 24 (5%) phrases were ambiguous in this way, and were coded as a single concept using my best judgment. These 24 concepts are included in the numbers cited in this section. The following graphic provides a complete breakdown of the concepts and the frequency of occurrence in this data set.

The findings from our initial coding provide a solid base for developing a more complete and reliable coding schema for problem descriptions. In the next section using

the 2010 Week 3 data set, we further develop the schema.

5.3.3 Phase 2: Refinement of SR.BID

Following the initial coding, I used two coders to refine and validate the SR.BID schema using the Week 3 2010 data set, which consisted of 37 design problem statements between one and two pages in length. I was the first coder, while the second coder was a third year undergraduate biology student new to the field of biologically inspired design, and without prior background knowledge in design or cognition, SBF or SR.BID. I allocated half of the data (17 problem statements, selected at random) to training and refinement and used the remaining to draw samples for testing and validation. Training and refinement occurred in sessions of 1-3 hours, one or two days a week, for approximately fifteen weeks; we estimate the total time spent at 80 hours for each coder. The entire training set was over 4300 words parsed into more than 1000 concepts, over a wide variety of problem types.

During this phase we refined SR.BID by generating additional sub-categories, which we added retrospectively based on my experience and reflection over the previous 2009 data set, and incrementally after instances of a new sub-category became evident from the 2010 refinement and training data set. Figure 5-10 shows the high level conceptual categories and their relationships. Appendix E provides a complete listing and description of each category and sub-category. The “component” category became a sub-type under constraint, which was expanded to “constraint or specification”. The inclusion of “specification” to the category title was to accommodate for phrases which expanded the design space such as “we could use a metal foam here,” and were specification oriented (that is, were about the device itself), but non-constraining.

Secondly, we pruned SBF category-seed types that were either infrequently used or presented inconsistencies in grain size. For instance, the key SBF concept behavior presents itself as a causally connected set of lower-grain size categories often distributed over a document; while a potentially useful category, its inclusion would require yet another level of abstraction over the schema (along with a complete evaluation of the set of new categories at that level of abstraction). I reason that since behavior accounted as fewer than 3 percent of the instances of concepts, the benefit of its inclusion was not warranted.¹²

In addition to the categorization of each concept, relationships among concepts were also identified and coded. These included type/sub-type relationships (A is a kind of/part of B), descriptive relationships (A describes, elaborates or places a value on B), qualifying relationships (A qualifies/constrains B). Through analysis of the 2009 data and incremental addition during the training and refinement step we generated a relationship matrix.

¹² The infrequency of the observation of behavior concepts may be an effect of context limitations, in particular an effect of the expertise of the observed design teams. Studies in the context of biologically inspired design using expert designers may find that an additional level of abstraction is required for richer analytics.

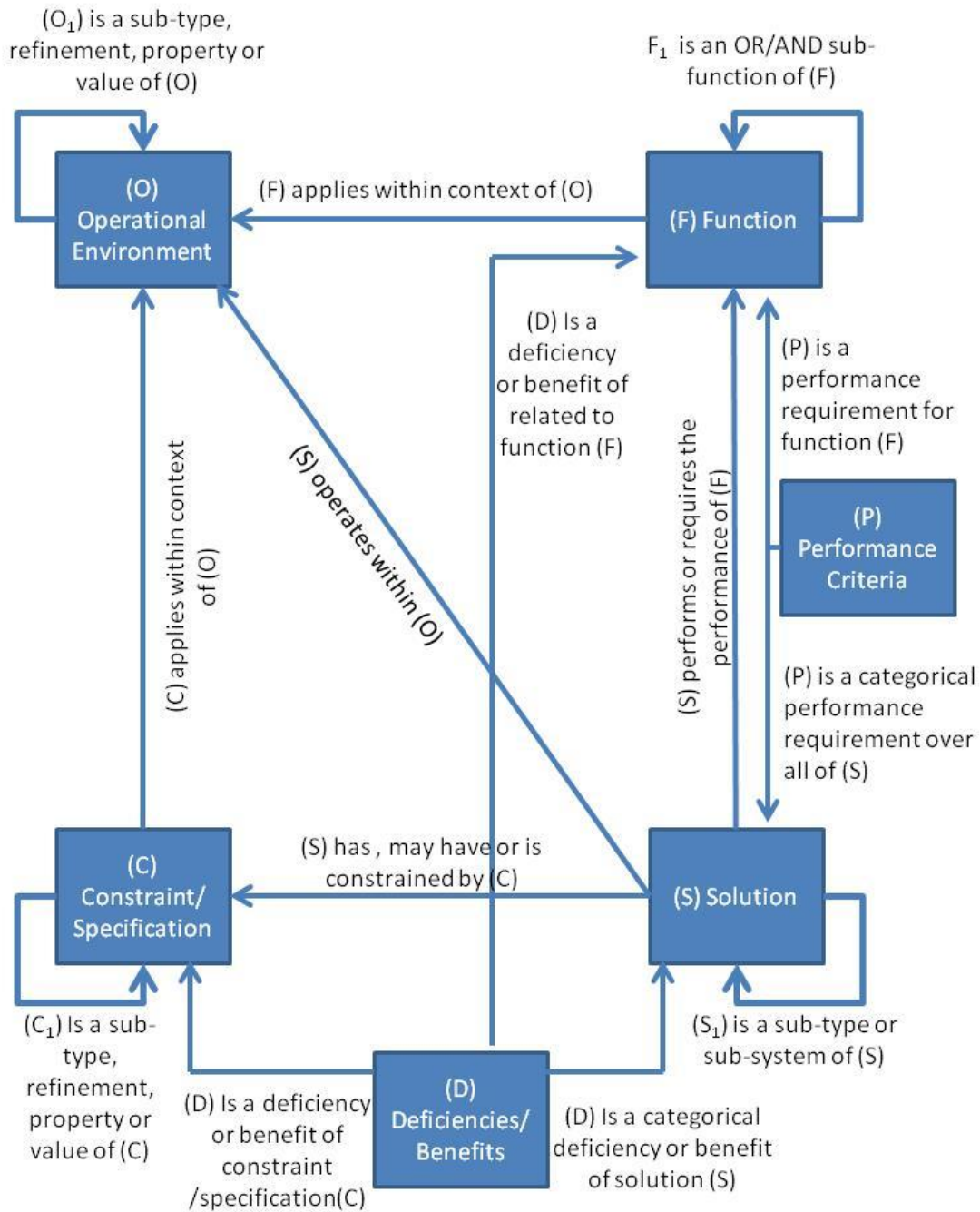


Figure 5-2: The primary conceptual categories and relationships of the SR.BID content account for problem formulation in BID.

After two passes on refinement and training, a random sample of five problem definition assignments were pulled from the data set to be used for validation. Each coder independently coded each test sample. A total of 246 base concepts were identified as relevant by both coders, 198 (80.5%) of which were in agreement. Using Cohen's Kappa as a measure of inter-coder reliability which adjusts for chance agreement, we arrive at a value of .778. Generally Cohen's Kappa values near 80% are deemed acceptable. Relationship concepts were fewer, 112, of which 84 (75.0%) were in agreement. The Cohen's Kappa value for relationships was .703, slightly less than is desirable. After initial comparison, the coders conducted a negotiation phase, in which they attempted to resolve coding discrepancies. As expected, post-negotiation agreement levels were significantly higher, 96.7% for concepts, and 98.0% for relationships with Cohen's Kappa values of .962 and .976 respectively.

Finally, the number of relevant concepts for which a code could be assigned post-negotiation was also studied, to ensure the schema provided comprehensive coverage of the domain. I analyzed the total number of "unknown" coded concepts versus the total number of concepts, arriving at a ratio of unknown to total concepts. This ratio was used as a simple measure of comprehensiveness.

5.3.4 Phase 3: Analytical Application of SR.BID

To test the conceptual soundness and potential usefulness of SR.BID, I applied it to the 2010 Week 8 data set, consisting of 31 brief problem statements. While previous validation tests confirm that I can achieve acceptable reliability with a single coder, because of the difficulty of coding relationship concepts, I use a more conservative dual-coding strategy over the entire data set. The complete rubric used by coders, which

depends on SR.BID but is a part of the method of coding, is available in Appendix F. During dual-coding, each of the two coders is present during the session, and while one coder takes the lead, the second coder may question coding decisions leading to discussion and negotiation until a code is agreed upon. This ensures reliability much closer to the post-negotiated numbers shown in the previous test, at the expense of requiring two-coders to code all documents. Dual-coding for all documents was conducted over 10 working sessions separated by at least 48 hours, lasting between 45 and 105 minutes each.

To validate the dual-coding strategy, after 16 weeks, the same coders applied the same methodology to 5 of the documents that had been previously coded and compared the results using Cohen's-Kappa. No additional negotiation was conducted. The number of "unknown" concepts was again analyzed as before to ensure comprehensiveness. The five random problem statements consisted of 164 concepts, 17 of which were considered not relevant to the design problem. The codes were then compared between the remaining concepts. Of the remaining 147 concepts, the coders matched on 129, or 87.8%, of the category assignments. Relationship coding was likewise compared, and matched on 63 of 72, or 87.2%, relationships. As noted previously, levels above 80% are usually deemed acceptable.

After coding, the 31 problem statements consisted of a total of 968 concepts, of which 112 were considered not relevant to the design problem. Of the 856 relevant concepts, 442 concepts were coded as relationships. The coders were unable to identify corresponding categories for 23 (2.7%) of the 856 concepts.

After coding, the 31 problem statements consisted of a total of 968 concepts, of which

Table 5-4. Non-weighted mean percentage, standard deviation and frequency of occurrence of each category.

	Standard		Frequency
	Mean	Deviation	
Functions	25.1%	9.1%	97%
Solutions	18.9%	11.3%	100%
Operating Environments	26.9%	15.7%	97%
Performance Criteria	5.3%	5.5%	61%
Deficiencies/Benefits	4.3%	5.4%	52%
Constraints/Specifications	5.6%	8.5%	42%

Table 5-5 Percentage breakdown by core concept and concept relationship

Core Concept	Related Concept	Number of Relationship Occurrences	Percent Relative to Number of Relationships
Operating Environment	Operating Environment	13	3.1%
Constraint/Specification	Constraint/Specification	0	0.0%
Solution	Solution	39	9.2%
	Operating Environment	5	1.2%
	Function	109	25.8%
	Constraint/Specification	47	11.1%
Function	Function	41	9.7%
	Operating Environment	66	15.6%
Performance Criteria	Function	67	15.9%
	Solution	4	0.9%
Benefit	Function	8	1.9%
	Solution	2	0.5%
Deficiency	Function	16	3.8%
	Solution	5	1.2%

112 were considered not relevant to the design problem. Of the remaining 856, the coders were unable to identify corresponding categories for 23 (2.7%) concepts. Table 5-4 shows the non-weighted mean percentage occurrence of each category in a problem

statement, the standard deviation, and the frequency with which the category occurs at all (i.e. solutions are always found, while deficiencies/benefits occur in roughly half of the problem descriptions). Of the 856 relevant concepts, 442 concepts were coded with relationships. Table 5-5 provides a percentage breakdown by core concept and the concept to which it is related.

Figure 5-11 provides a visual depiction of a somewhat short, coded problem description from our data set, which has been modified slightly from its original form for example purposes.

The development of the electric car is a great thing for car owners and the environment, since tail pipe emissions can be reused to zero, have less moving parts, and there have been huge developments in electric motors. However there is a problem in charging the battery. The time it takes to charge the battery is at least six hours. And there is limited range of the vehicle. There is a huge future for electric cars but electricity will still need to be generated to power them. The design problem is that it takes too long to charge.

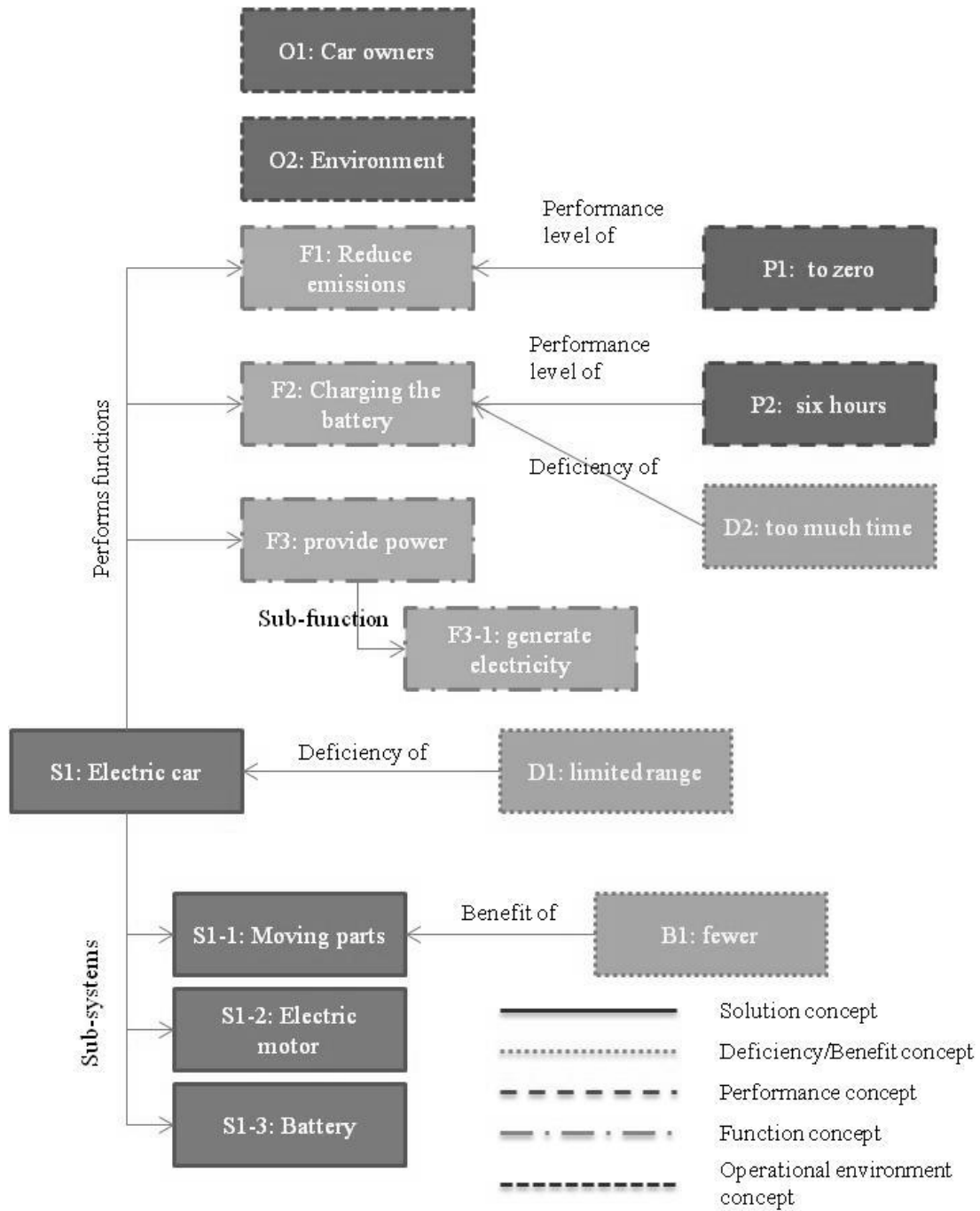


Figure 5-3. Sample SR.BID problem model.

My goal in presenting this data is to demonstrate the utility of the coding scheme. For example, table 5-4 shows the average use of function, yet with some variance (mean 25.1%, standard deviation of 9.1%). Such measurable variance among problem formulations raises questions about what makes for a “good” initial problem formulation, how much one should focus on function versus other aspects of the formulation desirable, and what effect one type of formulation versus another might have on both process and outcome. These questions can be extended to consider categorical composition, as well as number of type of relationships present.

When looking at the relationships in aggregate in table 5-5, functions appear in 72.7% of the relationships, suggesting they are deeply integrated with other concepts in the problem structure. This raises further questions, such as in what ways does expression of the connectivity of functions to other aspect of the problem influence the continued development of the problem formulation, the design process, and the final outcome?

Like function concepts, solution concepts are pervasive, and play a key role in organizing the problem structure, appearing in 50% of the relationships in table 5-5. The relative percentage of references to solutions is more variable than function (mean 18.9%, standard deviation 11.3%). We see, for example, that the solution-function relationship is most common, representing more than one-fourth of the conceptual relationships in the observed sample; nearly 70% of which are expressed as relationships between *existing* solutions and functions. Further analysis reveals that all of the constraints or specifications such as cost, components, materials, or properties are provided in the context of or relative to an *existing* solution (40/47), or less frequently, a *conjectured* solution (7/47). Understanding the role and influence of solutions is of particular interest

in the domain of biologically inspired design. By looking into the relationships between solutions and other concepts in the problem description, we may begin to shed some light on how biology influences not only solution outcomes, but also problem formulation.

The constraints and specifications category accounted for only 5.6% of the concepts on average. Moreover, only 42% of the problem descriptions had constraint/specification concepts at all. This may be a result of the stage of the design cycle in which the data was collected; the innovative, open-nature of the classroom context; or the fact that there were no customers involved in the problem formulation. By considering the composition of problem formulations in this way, we may reveal context differences that influence problem formulations which in turn may be able to explain variances in solution outcomes.

5.4 Summary of SR.BID

In this section I used the method of ontologically grounded theory to build the SR.BID schema for analyzing the relationship between biological analogues and problem formulation in the context of the biologically inspired design classroom. I justified the use of the Structure-Behavior-Function ontology as a seed ontology, from which to build the SR.BID schema. I mapped the SBF schema to a first set of data and expanding high-level categories to account for disparities between the schema and observations. In combination with a second, coder unbiased with prior knowledge of SBF or the domain of biologically inspired design, I generated a robust categorization schema, SR.BID, for categorizing concepts and conceptual relationships in problem formulations.

Recall that the criteria established at the outset for the evaluation of the content account are: (1) the ability of the account to reliably attain the same results for a given

analogue/problem, (2) the ability of the system to attain results across the range of biological analogues and mechanical engineer design problem observed in class, and (3) the ability of the account to enable analysis of the correlations between design problems and biological analogues.

Using this second set of data I showed that SR.BID was both a comprehensive and reliable schema for coding problem descriptions, accounting for the first two criteria. I account for the third criteria, through the coding and subsequent analysis of 31 additional problem descriptions, by which I demonstrated the utility of SR.BID for performing an aggregate analysis of problem descriptions. This leads to the following claim:

5.5 Claim 2

I claim that SR.BID provides a comprehensive and reliable account for representing problem descriptions, biological analogues and the relationship between them.

In Chapter 6 I will address the second research question. In that chapter I will use the SR.BID framework to first scaffold the understanding of analogical problem evolution (APE), and then to help build a process framework in which to explain analogical problem evolution.

6 A PROCESS ACCOUNT OF PROBLEM EVOLUTION

Analogical problem evolution (APE) as a phenomenon lies at the intersection of two as yet not integrated aspects of design theory: analogical design and problem-solution coevolution. From the perspective of analogical design theories, APE exhibits the retrieval-mapping-transfer behaviors one would expect from those theories. However, in traditional theories of analogical design, especially theories of analogical design as applied to BID, transfer occurs between a biological *source* and a conjectured design *solution*. APE exposes a new opportunity for the application of analogical theories of design to not only solution generation, but to problem evolution as well. In problem-solution coevolution a design problem evolves in response to the evaluation of a *conjectured* solution. Similarly in APE, the design problem evolves in response to a solution. Unlike in traditional problem-solution coevolution, however, in APE the problem evolves in response to an existing (analogue) solution from the domain of biology. APE provides an opportunity to extend existing problem-solution coevolution theories on one hand into cross-domain solutions and on the other hand into existing solutions, as a means to evolve problems.

6.1 Research Problem 2

Current theories of analogy account for analogical evolution of solutions; while current theories of problem evolution account for problem evolution in the context of conjectured solutions. Analogical problem evolution remains unaccounted for in either analogical design theory or problem-solution co-evolution design theory.

As a first step toward a process account, I require a better description of the process of APE. Until now, descriptions of APE have been inferred from high level task-level process accounts of related phenomena, such as solution-based design and compound analogy, or from observational accounts such as the design trajectory accounts generated from the 2006 and 2007 studies. In order to develop a process account of APE, I first require an account of *what* and *when* content is transferred from existing biological solutions to design problem formulations. *The SR.BID framework and coding scheme provide a means for generating this more detailed understanding of the interactions between problem and analogy in the APE process.* The first hypotheses in this section deal with the development of a theory of problem evolution (PE.BID) within which APE occurs.

6.2 Study P.1: Qualitative Analysis of APE using SR.BID

6.2.1 Study Motivation

This study is motivated by the need to further characterize *what* is transferred and *when* transfer occurs in the process of analogical problem evolution.

6.2.2 Methodology

6.2.2.1 Research Question P.1

What is the content is transferred from biological analogues to problem formulations, and *when* is it transferred in APE?

6.2.2.2 Hypothesis P.1

An encoding of design problem formulations in terms of SR.BID will demonstrate the content transferred between biological solutions and design problems in the process of

BID.

6.2.2.3 *Context and Participants P.1*

This study uses one design trajectory case from 2009, called the SolShield project. Data was collected in the context of the BID@GATech classroom in 2009. See section 3.2 and 3.3 for a complete description of the 2009 classroom context.

6.2.2.4 *Materials P.1*

This study uses data collected from individual problem description assignments, a team midterm presentation and a team final design report assignment. The details for the assignments used as prompts for students are documented in section 3.3.3 and 3.3.4.

The design trajectory materials are the same those previously analyzed in Phase 1 of the development of the SR.BID (see section 5.4.2). This design trajectory was selected because it was a known example of analogical problem evolution. A complete description of this design trajectory was also provided in section 3.2.4 as an illustrative example of biologically inspired design.

6.2.2.5 *Execution of Study P.1*

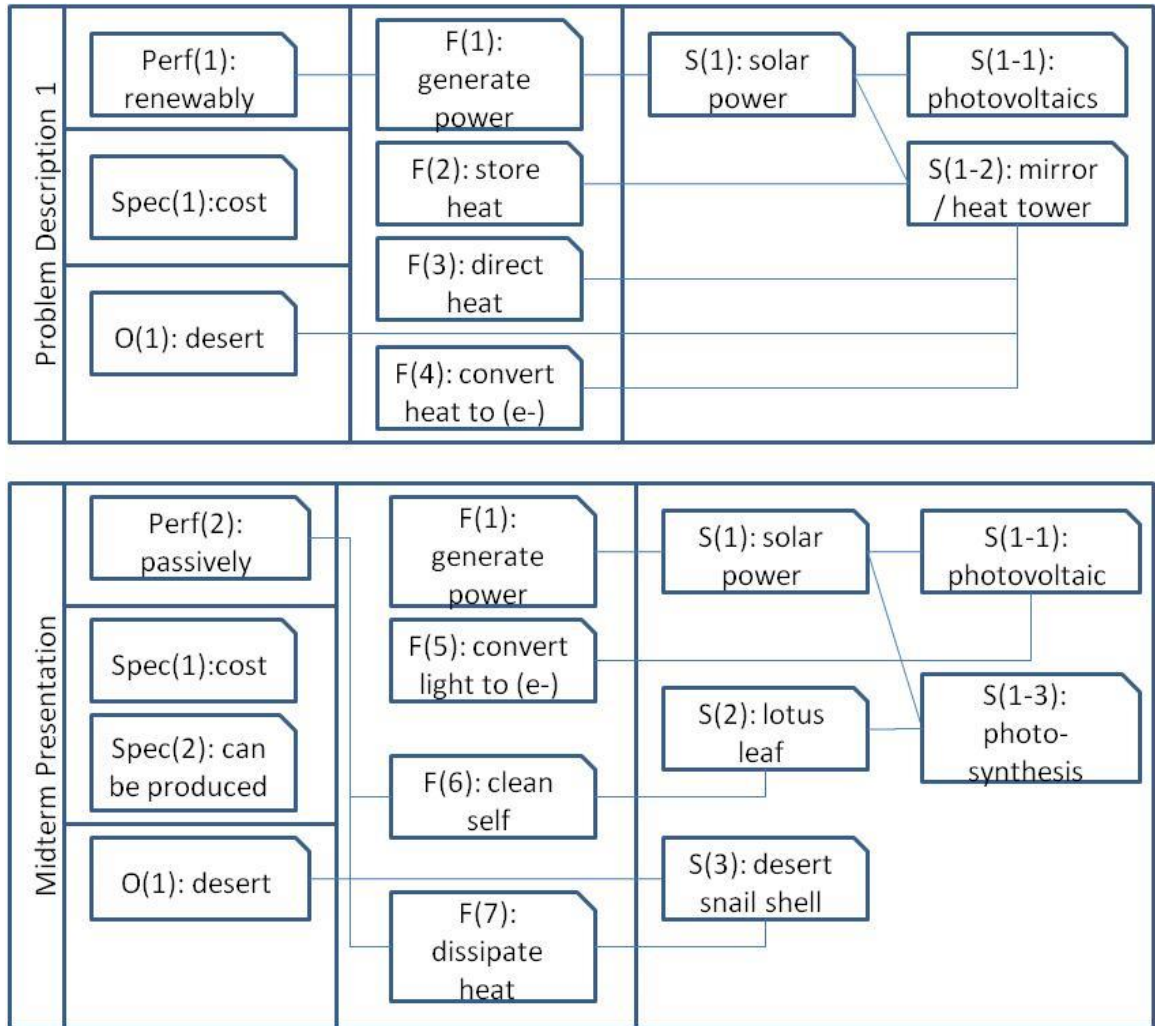
Using the SR.BID content account of problem formulation in BID, I encode the design materials collected in the SolShield project. The materials include point in time, self-generated descriptions of problem formulations, biological analogues, and solutions generated during the semester long course of the design. I use the methodology developed and tested in section 5.4 of the previous study. I construct from these encodings for each point in time a design problem model. I then perform a differential analysis across models to identify which concepts and relationships were added, deleted, or carried over from previous models. Conceptual transfer from biological analogues to

problem conceptualization is inferred through relationship encodings between new concepts and biological solutions.

6.2.2.6 *The Technique of Differential Analysis*

Differential analysis of two problem models involves identifying the similarities and differences between design problem models over time. Models encoded using SR.BID, each representing a design problem description at a point in time, can be compared at the concept and relationship levels. For models collected as complete descriptions at discrete points in time (as in this study), the individual concepts and relationships from model at time t are compared to previous models from time $t-1$, $t-2$, etc. Each concept or relationship is then encoded as either ***dropped*** (in $t-1$, but not t), ***new*** (in t , but not $t-1$), existing from the ***prior*** model (in both t and $t-1$), or ***re-introduced*** (in t , not in $t-1$, and in $t-2$ or prior).

Take for example the partial models taken from problem description 1 and the midterm problem description, shown in Figure 6-1. Table 6-1 and table 6-2 represent the differential encoding. Table 6-1 represents the concept differential. Table 6-2 represents the relationship differential.



Key

Perf = performance criteria, F = function, O = operational environment, S = solution
 Spec = specification

Figure 6-1. Two partial models of SolShield design problem from adjacent time steps.

Table 6-1. Example differential analysis of concepts between problem models.

Problem Decomposition 1		Midterm		Concept
Category	Concept	Category	Concept	Differential
perf(1)	renewability			dropped
		perf(2)	passively	new
spec(1)	cost	spec(1)	cost	existing
o(1)	desert	o(1)	desert	existing
f(1)	generate power	f(1)	generate power	existing
f(2)	store heat			dropped
f(3)	direct heat			dropped
f(4)	convert heat to e-			dropped
		f(5)	convert light to e-	new
		f(6)	clean self	new
		f(7)	dissipate heat	new
s(1)	solar power	s(1)	solar power	existing
s(1-1)	photovoltaics	s(1-1)	photovoltaics	existing
s(1-2)	mirror/heat tower			dropped
		s(1-3)	photosynthesis	new
		s(2)	lotus leaf	new
		s(3)	desert snail	new

Table 6-2 Example differential analysis of relationships between problem models.

Problem Decomposition 1		Midterm		Relationship
Concept 1	Concept 2	Concept 1	Concept 2	Differential
f(1)	Perf(1)			dropped
f(1)	s(1)	f(1)	s(1)	existing
f(2)	s(1-2)			dropped
f(3)	s(1-2)			dropped
f(4)	s(1-2)			dropped
		f(5)	s(1-1)	new
		f(6)	s(2)	new
		f(6)	perf(2)	new
		f(7)	s(3)	new
		f(7)	perf(2)	new
s(1)	s(1-1)	s(1)	s(1-1)	existing
s(1)	s(1-2)			dropped
		s(1)	s(1-3)	new
		s(2)	s(1-3)	new
o(1)	s(1-2)			dropped
		o(1)	s(3)	new

The differential encoding allows the tracking of concepts over time. In looking at analogical problem evolution, a differential analysis can (a) determine when a new concept is added, and (b) determine the relationship that concept has with other concepts, in particular, those concepts related to biological solutions (such as the lotus leaf or desert snail). This provides the ability to describe *what* concepts from biological analogies are added to the problem description. The timing of conceptual transfer is implied as between problem snapshots. In the case of this study, snapshots are gathered roughly every three weeks. This provides the ability to describe *when* concepts from biological analogies are added. The level of granularity with respect to time describes a three week window.

6.2.2.7 Data P.1

A sample of the data collected, and a sample of the encoding of the first two

paragraphs of that data is included here. This sample represents the first two paragraphs of the second team problem decomposition from the SolShield team. A description of this design trajectory was also provided in section 3.2.4.

Problem definition document text:

I think that our charette went over pretty well and that we got some wonderful feedback that will help give us new insights and improve on our project. A problem statement is currently to improve solar design for structural uses by increasing efficiency, introducing sustainable materials, and making it dynamic. I think that our 5 analogies of the lily pad, diatoms, negative feedback, snail shells, and photosynthesis provided a wide range of possibilities to accomplish our current mission statement, but further decomposition is needed.

Solar panels would be more viable as an energy source if they can become better at conserving energy. This involves them becoming more efficient and would ultimately be more beneficial if they used more sustainable materials. Efficiency can be improved by making these structures more dynamic and respond to the environment. Another aspect of conserving energy is to make the transmission of energy more conserved, so that the amount of energy generated in a solar panel is the same transmitted to the energy receptors. Dealing with the sun, it is also important in energy conservation to retain energy when it is not sunny, or to initiate another process that will generate energy.

The text is transcribed into a spreadsheet, and decomposed into individual concepts. Each concept is then encoded in terms of a primary categorical encoding, one or more sub-category encodings, and a relationship encoding, per the SR.BID encoding rubric in Appendix F. The complete design document and encoding is available in Appendix G. Notes are added at the time of encoding to capture the interpretation of the coder. The column heading “comment” indicates that the text refers to issues not directly related to the design problem or solution. These are often comments about the designers themselves, their design processes, design challenges they are facing, or references to sources of information. Table 6-3 provides a encoding of the first two paragraphs of the text.

Table 6-3 Example SR.BID encoding of two paragraphs of text from a design problem definition.

	Statement	Notes	Comment	Primary Concept	Relationship
1	I think that our charette went over pretty well and that we got some wonderful feedback that will help us give new insights and improve on our project.		Comment		
2	A problem statement is currently		Comment		
	to improve	a requirement on one or more of the functions F(any) t.b.d., of S(0)		perf(1): improve on (existing)	s(0): solar panel
3	solar design	the design of a solar panel, an existing solution S(0)		s(0): solar panel	
4	for structural uses	operational environment of S(0) - solar panels on stationary structures (e.g. roofs, ground, etc)		oe(1, 1): on statonary structures	
5	by increasing efficiency	is a comparative requirement on improving solar design		perf(2): more efficiently	s(0): solar panel
6	introducing sustainable materials			spec(1, mat): sustainable materials	s(0): solar panel
7	and making it dynamic.	it = the system, new sub of F(1)		f(1-1): make (self) dynamic	s(0): solar panel
8	I think that our five analogies of the		Comment		
9	lily pad,	specific solution		b(1): lily pad	
10	dynamic feedback,	general pattern		b(2): dynamic feedback	
11	snail shells,	specific solution		b(3): snail shells	
12	and photosynthesis	general process		b(4): phtotsynthesis	

Table 6-3 continued

13	provided a wide range of possibilities to accomplish our current mission statement but further decomposition is necessary.		Comment		
14	Solar panels would be more viable as an energy source	amplification of F(1)		f(1): generate solar energy	s(0): solar panel
15	if they can become better at conserving energy	new sub of F(1)		f(1-2): conserve energy	s(0): solar panel
16	this involves them becoming more efficient			perf(2): more efficiently	s(0): solar panel
17	and would ultimately be more beneficial if the used more sustainable materials.			spec(1, mat): sustainable materials	s(0): solar panel
18	efficiency can be improved by making these structures more dynamic			f(1-1): make (self) dynamic	s(0): solar panel
19	and respond to the environment	new sub of F(1-1), responding to the environment is a way of making the system more dynamic		f(1-1-1): respond to environment	s(0): solar panel
20	another aspect of conserving energy	the word "another" suggests a connection between conserving energy F(1-2) and efficiency F(1-1) in the previous statement		f(1-2): conserve energy	s(0): solar panel
21	is to make the transmission of energy	new sub of F(1-2)		f(1-2-1): transmit energy	
22	more conserved	is a comparative requirement on the transmission of energy		perf(3): more conserved	f(1-2-1): transmit energy
23	so that the amount of energy generated in a solar panel	amplification of F(1)		f(1): generate solar energy	s(0): solar panel

Table 6-3 continued

24	is the same transmitted	amplifies the comparative requirement on the transmission of energy		perf(3): more conserved	f(1-2-1): transmit energy
25	to the energy receptors.	in the operational environment there are energy receptors		oe(2, u): receivers of energy	
26	dealing with the sun	the sun is part of the operational environment		oe(3, c): sun	
27	it is also important in energy conservation	related to F(1-2)		f(1-2): conserve energy	s(0): solar panel
28	to retain energy	new sub-function of F(1-2)		f(1-2-2): retain energy	
29	when it is not sunny	condition constraint: sun availability		oe(4, c): not sunny	f(1-2-2): retain energy, f(2): initiate another process
30	or to initiate another process	a function parallel to F(1),		f(2): initiate another process	
31	that will generate energy.	new super-function of F(1) generate solar and F(2) generate non-solar energy	super to f(1)	fs(1): generate energy	

These spreadsheets are transformed by hand into graphical model representations. Figure 5-11 and figure 6-1 show small graphical representations of models. These models consist of dozens of concepts and relationships, summarizing the encoded representation and make certain patterns of connectivity obvious, for example functional decompositions and highly connected nodes. The graphical representations allow visual comparisons, and simplify the encoding by eliminating redundancy in the encoding.

6.2.2.8 *Evaluation P.1*

Documents are encoded using the SR.BID encoding rubric and encoding process developed in study C.3. The data used in this study are the same data used in the Phase 1 Initial Coding of this study, where they were used to understand the degree to which the SBF conceptual framework was able to encode problem formulations. At that time, they formed a starting place for the development of the complete SR.BID. In this analysis, I re-encode the same case using the now complete SR.BID encoding scheme¹³.

The encoding of each problem formulation provides the ability to quantify concepts in number and kind for each problem formulation. This analysis stands in contrast to the previous analysis in phase 3 of study C.3 in one important way. In that analysis problem formulations were quantified for a *single point in time* across a number of *different projects*. In this analysis I will provide a quantification of concepts and relationships for a single project description across multiple points in time. This will reveal what concepts in the problem formulation change and when they change across the design trajectory.

¹³ The encoding scheme and process was previously validated using data distinct from this case study, as described in section 5.4.4.

6.2.3 Results of the Study

Table 6-4 provides a count of concepts for the four data points, across high-level SR.BID categories. I separate solution concepts into “existing” and “design” solution concepts, where “existing” solutions include both biological and man-made solutions and “design” solutions are new design solutions conjectured by the design team. Only unique concepts are counted; repetitions of the same or very similar concepts are not counted. At this level of analysis, we see the number of functions considered at each stage remains between 20 and 25 until the final design, where it drops to 9. The number of performance criteria, 25% of which were with respect to efficiency, was consistently high (between 10 and 17) for the first three data points, and decreases to only a few (3) at the final presentation.

Table 6-4, Number of concepts, by concept type across four points in time.

	PD1	Midterm	PD2	Final
Operational environment	23	4	6	12
Function	25	20	20	9
Constraint/Specification	1	6	3	4
Performance criteria	10	10	17	3
Existing solutions	9	8	6	5
Design solutions	1	7	3	1

This parallels the reduction in functions considered in the final presentation. The number of constraints and specifications is very low, never more than 6. Of the 14 total, four were cost-related and three were sustainable materials related. The number of operational environment concepts initially considered was very large (23), but was rapidly reduced (to 4) by the second data point, and then gradually expanded throughout the remainder of the design. It appears that the designers ranged across many different

environments initially, but then settled into a single environment (household mounted solar panels), which gradually took on finer levels of detail. Finally, the number of existing solutions references was consistent, between 5 and 9, trending down to 5 at the final presentation. The number of new solutions discussed moves from one (literally, “the new solution”), to seven – an exploration of independent solution ideas – back to a single final new solution in the end.

Table 6-5. Function concept carry through from problem descriptions.

	Present in PD1	Present in Midterm	Present in PD2	Present in Final
Added in PD1	25	3	7	2
Added in Midterm	--	17	3	1
Added in PD2	--	--	10	1
Added in Final	--	--	--	5
Total	25	20	20	9

With respect to functions, Table 6-5 uses differential analysis to identify functions which were added one problem iteration and were present in later iterations. The table is interpreted as follows, the rows represent when a function was added, and the columns represent the number of functions from that iteration that were present in a given problem description. Reading across the first row of the table (PD1), 25 new functions were added in PD1, which is also the total number of functions (carried to the bottom row). Continuing across to the next column, the number of functions from the initial formulation of 25 that are carried over in the midterm is 3. Likewise 7 of the initial functions are found in PD2, and 2 of the initially proposed functions are found in the final description. That is, only 2 out of 25 of the initially proposed functions are found in the final problem description (“generate energy” and “capture energy”). The second row shows the number of new functions that were added, and subsequently carried over from

the Midterm, likewise the row below for PD2.

In the final problem description itself, there are more new functions than old. Five new functions appear in the model, while four have been carried through from previous descriptions. This alone tells a very interesting story. In this ill-defined design problem, we see a great deal of exploration. Fifty-seven unique functions are considered, only nine of which are described in the final solution. Over 80% of the functions considered are discarded along the way.

Next I explore the evolution of the problem *relative to analogies* by examining the relationships of existing solutions to the concepts in the problem model. For any concept in the problem model, that concept may be associated with an existing solution or with a new solution concept. The differential analysis shows when and where such relationships are added, relative to an existing concept. The following four tables show for each data point the numbers of operational environment, function, artifact specifications, and criteria concepts respectively and whether they are associated with (1) an existing solution, (2) a biological solution, or (3) no solution association. I will discuss each table in order. I note that the numbers in these four tables will be equal to or greater than the total number of concepts reported for each category, as a concept may be associated with one or more solution category. For example if both a biological and a non-biological solution were related respect to a function, we would get two tallies for that concept. Likewise multiple solutions in the same category could reference the same concept.

Table 6-6. Operational environment concepts by reference to source.

	PD1	Midterm	PD2	Final
Man-made	9	0	1	9
Biological	0	5	0	0
No Reference	15	0	5	10
Total	24	5	6	19

Table 6-6 shows that many operational environment concepts were initially generated from existing solutions (9), but even more concepts (15) were not associated with any solution. We see the opposite occur in the midterm, where the only operational environment concepts were associated with biological solutions, such as “desert areas” and “the presence of the sun” where we find the desert snail. Interestingly, after this short period of mentioning biological source operational environments, we don’t see a single reference to them for the remainder of the design process. The remaining operational environments are either related to man-made solutions, or are not associated with any particular solution.

Table 6-7. Function concepts by reference to source.

	PD1	Midterm	PD2	Final
Man-made	18	4	10	4
Biological	0	12	6	6
No reference	11	6	12	3
Total	29	22	28	13

The trends for function in table 6-7 somewhat mirror those for operational environment across the first two data points, but whereas the operating environments from biological sources are dropped in the 3rd iteration, biological functions are “stickier”, eventually comprising almost half (6 of 13) of the functions mentioned in the final presentation. Of the three new functions introduced in the final design, all were

taken from biological sources.

Of the 14 specifications mentioned, none were associated specifically with other solutions. Most of the specifications were with regard to cost and sustainable materials, which were likely inferred from the design context of “sustainable housing”, or from sustainability values expressed by the instructors (note, there were no “customers” per se with whom designers could interact).

Table 6-8. Performance criteria by reference to source.

	PD1	Midterm	PD2	Final
Man-made	0	2	11	1
Biological	0	3	2	1
No reference	10	5	4	1
Total	10	10	17	3

Table 6-8 shows that none of the 10 performance criteria in the initial problem formulation are grounded to a man-made or biological system. Initial performance criteria were vague, for example “to design a more efficient solution”. While “more efficient” is comparative, designers did not provide a system against which to compare. This trend reverses itself when designers have narrowed the problem scope, and by the third data point 13 of 17 references are now grounded to other solutions. Of these only a few are with respect to biological solutions, suggesting that performance criteria are established against other man-made products, not against biology.¹⁴ In the final design, performance criteria practically disappear altogether. This may be a result of the rapid pruning of their final solution to a few functions in the final step of the process, and

¹⁴ If performance benchmarks are typically against man-made products, even in biologically inspired design, what does this say about historical trends in sustainability? Perhaps one of the benefits of biologically inspired design is in the consideration of other standards of performance.

narrow scoping for their detailed quantified analysis.

6.2.4 Summary of Study

This analysis provides a breadth of additional information for consideration in a process model of APE.

- In the context of this biologically inspired design class, the number of total function concepts explored is roughly 5 times greater with respect to functions than the number of function concepts in the final design. Over 80% of the functions explored are discarded by the final design phase.
- Either existing man-made, existing biological solutions or both are cited with respect to the formulation of the problem; this occurs in every problem formulation thus far observed. Design problem formulation and existing solutions appear to be deeply connected, at least during conceptual design.
- Concepts from biological solutions are not used in the *initial* problem formulations; rather existing man-made solutions provide the basis for many of the concepts in the initial formulation. Concepts from biological solutions are integrated only in later stages.
- Certain conceptual categories in problem formulation are transferred more frequently from biological analogues than others. For example, while 38% of functional concepts appear to be transferred from biological analogues, only 20% of performance criteria are found in common, only 17% of operational environments, and no specifications/constraints appear in common.

I follow this analysis with a conjectured process account of problem evolution for

biologically inspired design, called the *problem evolution for BID* (PE.BID) model. I will break the model into six components, each of which will be assessed independently.

6.3 Study P.2: Developing a Process Account of APE

6.3.1 Study Motivation

This study is motivated by the need to develop a process account of analogical problem evolution. While existing theories of design account for problem evolution in response to conjectured solutions, in particular in response to the evaluation and failure of conjectured solutions, analogical problem evolution occurs outside of the conjecture-evaluation design cycle. Moreover, while existing theories of analogical design as applied to biologically inspired design account for the generation of new design solutions by analogy, they do not consider the generation of new problems by analogy. The previous study for a single design case provides a description of **what** is transferred in APE, and **when** it is transferred. This study conjectures a process account of **how** and **why** analogical problem evolution occurs, providing enough detail and support for underlying processes to inform future research.

6.3.2 Methodology

In this section, I first develop a general process account for *problem evolution* in biologically inspired design. This development of this process account is informed by previous research, classroom observations and experience. I will decompose and examine the process account as a number of sub-processes, investigating each sub-process to the

degree to which existing data and research permit. Framed in terms of this general account of problem evolution, I will then describe **how** and **why** analogies may be used in key aspects of the process of problem evolution.

6.3.2.1 *Research Question P.2*

What is a process account of design provides an explanation for APE in the context of problem evolution?

6.3.2.2 *Hypothesis P.2*

Cases of analogical problem evolution can be explained in terms of the PE.BID model of process evolution.

6.3.2.3 *Execution of Study P.2*

I will first propose PE.BID, a process account of problem evolution. Figure 6-2 provides a graphical representation of the PE.BID theory. This account specifies the processes and underlying memory requirements for describing **why** and **how** problem evolution occurs. I decompose PE.BID into sub-processes, and describe and investigate each of the components of this theory in turn, using an abbreviated format, touching on the key research question, hypothesis, method, evaluation and results for each.

6.3.2.4 *The PE.BID Process Account*

Design problems (P_1 , P_2) represent the design problem model. The SR.BID content account provides an ontology for representing such a model in memory. The design problem P_1 represents the model at a point in time; the design problem P_2 represents the same problem after it has evolved.

Designer problem goals (DPGs) translate a high-level design motivation into a design problem goal for the transformation of the problem. It is important to distinguish between

the goal with respect to changing the problem and the motivation for such a goal. I assume that the designer desires to change the problem formulation due to some external or internal motivation.

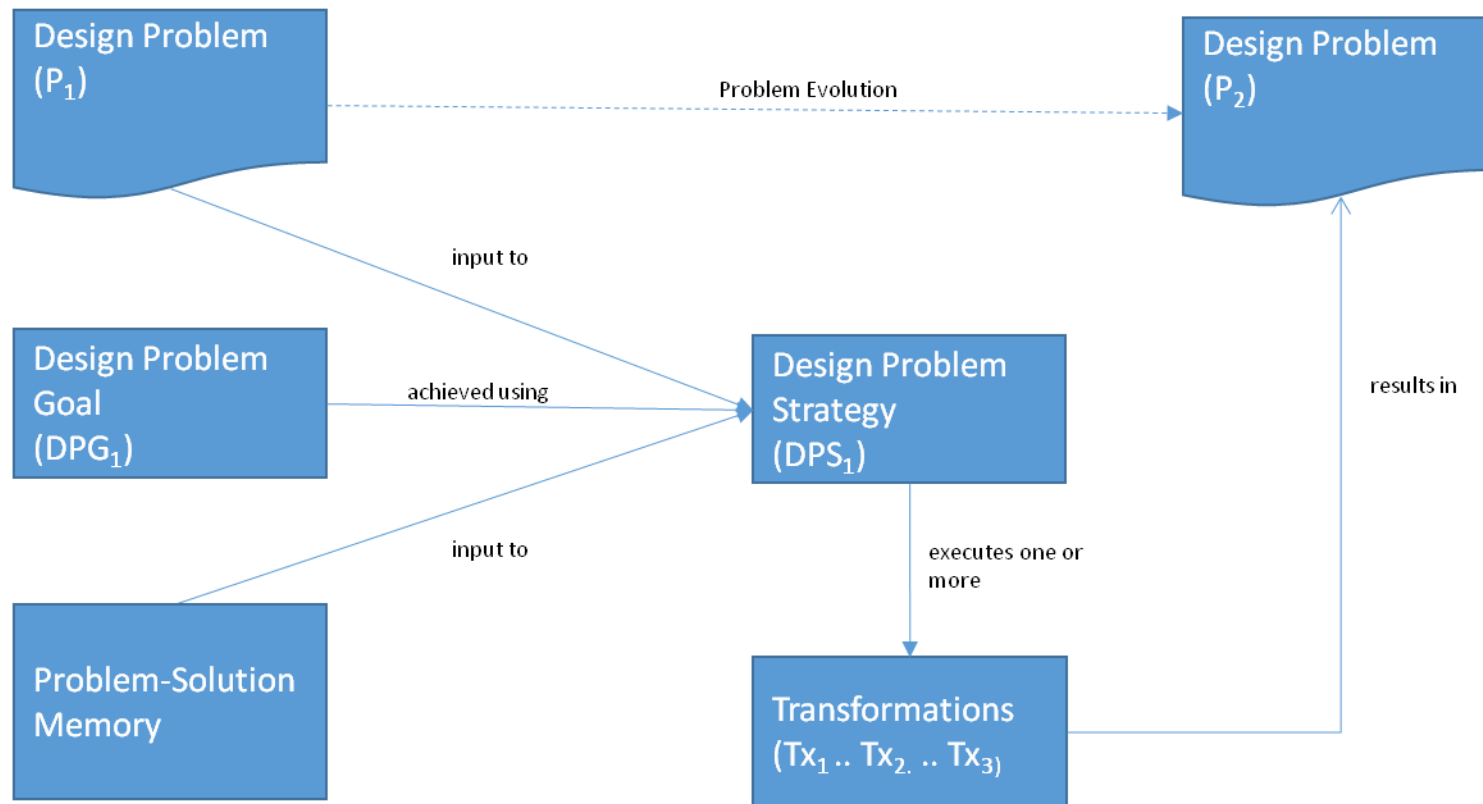


Figure 6-2: Graphical representation of the PE.BID theory of problem evolution in biologically inspired design.

While the exploration of designer motivation is beyond the scope of this work, some external motivations observed in the context of the BID class include responding to or resolving: a design flaw, sub-optimal performance, simulation failure, instructor/expert feedback, designer communication problems, and design team conflict resolution. Internal motivations may include: desire to be creative, curiosity, resolving ambiguity, meeting instructor expectations, the desire to impress, etc. Any of these motivations may trigger a designer to reconsider their problem formulation, and there may be correlations between certain motivations and certain problem goals (for example, motivation to resolve simulation failure may tend to generate goals for reducing design problem complexity).

Designer problem goals provide a set of general parameters for the resultant state of the transformation. For example, the goal of complexity reduction creates a resultant problem state with either (a) fewer concepts, or (b) with sub-problems each of which contains fewer concepts than the original problem.

Designer problem strategies (DPSs) provide the process by which the problem goal is accomplished. For example, if the goal is to reduce complexity, the designer may decompose a concept into one or more sub-concepts, each of which the designer can then consider independently (at the cost of increasing connectivity, which may create additional design challenges). Decomposition is a strategy (a DPS) to reduce complexity (a DPG). Each DPS specifies an initial state and resultant state of the design problem (which must meet the criterion established by the goal), a focal concept, and the knowledge requirements to execute the strategy.

Transformations provide the breadth of concept-level conditions and operators that a

strategy may access to achieve its goals. Transformations specify initial and resultant states of the design problem concepts. Strategies employ one or more transformations to achieve a designer problem goal.

The problem-solution memory provides the content of existing problems and solutions models, which may be required to execute particular strategy. For example, if the strategy specifies a functional decomposition, in addition to specifying which concept to decompose, the strategy requires knowledge about what sub-functions the concept may be decomposed into. The problem-solution memory provides a set of existing problem and solution model that provide this information.

In the remainder of this chapter, I will provide more detail and support for each aspect of the PE.BID model of problem evolution. This research is a work-in-progress. For each aspect, I will conjecture a hypothesis about that aspect of the process account, and provide a method of evaluation for that hypothesis. In those cases where studies are complete, I will provide results.

6.3.2.4.1 Hypothesis P.2.1: Designer Problem Goals

Table 6-9 provides the five characteristics of complex problems from Funke (2012) that ground the taxonomy of designer problem goals. Table 6-10 projects these goals into expected observable changes in the problem resulting state.

6.3.2.4.2 Method P.2.1

For design problem goals, I assume that design is a complex problem solving activity. According to (Funke 2001, 2003), complex problems exhibit five characteristics that simple problems do not. Funke proposes problem solver goals which address each of these characteristics. I hypothesize that designers generate similar problem goals to

resolve the difficulties that arise from these characteristics in design.

Table 6-9. The five characteristics of complex problems and associated goals.

Characteristic	Description (verbatim, Funke, 2012)	Goal
Intransparency	Intransparency concerns the variables involved and the definition of the goal. In an intransparent situation, not all required information about variables and possible goals are given.	Intransparency requires from the problem solver the active acquisition of information.
Complexity	Complexity is defined based on the number of variables (concepts) in the given system.	Complexity demands from the problem solver a simplification through reduction.
Connectivity	It is not the pure number of variables that is decisive for the workload on the problem-solving person, but the connectivity between these. Assuming that in a system of 100 variables every variable is connected to only exactly one other, the connectivity is lower than in a system in which all variables are connected to each other.	For making mutual dependencies understandable, a model of the connectivity is required from the problem solver.
Dynamics	This feature explains the fact that interventions into a complex, networked system might activate processes whose impact was possibly not intended. It signifies that in a lot of cases the problem does not wait for the problem-solving person and his/her decisions, but the situation changes itself over time.	Dynamics requires from the problem solver the consideration of the factor "time."
Polytely	Usually there is more than one goal in a complex situation that has to be considered. These goals may be in conflict.	Conflicts due to antagonistic goals require the forming of compromises and the definition of priorities.

6.3.2.4.3 Evaluation P.2.1

I assume that design in the context of the biologically inspired design course is a complex problem solving activity. I validate this by providing examples from common experience or from previous studies of each of the five characteristics of complex problem solving activity.

Table 6-10. Transformation resultant states

Characteristic	Resultant state characterization
Intransparency	The transformed problem provides greater information in terms of number of concepts and connections than the previous state.
Complexity	The transformed problem provides either (a) a problem with fewer concepts, or (b) one or more sub-problems with fewer concepts than the previous problem state.
Connectivity	The transformed problem provides either (a) a problem with fewer connections, or (b) one or more sub-problems with fewer connections, or (c) a problem with fewer average connections per node, than the previous problem state.
Dynamics	The transformed problem has more temporal/causal concepts or more connections to temporal/causal concepts.
Polytely	The transformed problem provides new or changes to existing prioritization criteria. ¹⁵

6.3.2.4.4 Results P.2.1

Table 6-11 provides a small set of examples from design literature or from observations in biologically inspired design, supporting the connection between complex problem solving characteristics and design.

Assuming that design is a complex problem solving activity that meets the five characteristics defined by Funke, then the goals for addressing these problem characteristics should align. There are many example behaviors from design instances that support this assumption. Table 6-12 shows observed example behaviors from BID or general design behavior in support of design as a complex problem solving activity.

¹⁵ Prioritization criteria were not emphasized in the context and rarely seen in the data. The current SR.BID content account defers the concepts of prioritization criteria to “design meta-problems” and not to the “design problem” per se, excluding it from the current definition.

Table 6-11. Design examples associated with complex problem characteristics.

Characteristic	Design example
Intransparency	Citing previous literature, Reitman (1964) highlights the under-specification of design problems. Dorst (2003) provides a categorization for design problems by the amount of indetermination. In modern design, that design problems are ill-defined is accepted as given; as Cross (2001) put it in his review of 30 years of design studies, “It is widely accepted that design ‘problems’ can only be regarded as a version of ill-defined problems.”
Complexity	The number of concepts expressed in a single problem formulation, as coded in SR.BID in the studies documented in section 5.4, ranges from small problems of 20 concepts to very large problems with over 100 concepts. Considering these are the expressed concepts, and each expressed concept may be supported by one or more related unexpressed concepts, the number of concepts contained in a problem model may be very large.
Connectivity	In addition to complexity in the number of concepts, the number of relationships explicitly expressed in an SR.BID problem model are about half of the number of concepts expressed, or between 10 and 50 relationships expressed per design problem.
Dynamics	The problem of dynamics is coupled with the idea that a conceptual design is intended to be embodied in an environment that can, and will, change. When designers consider safety factors and operational ranges, they are explicitly accounting for changes in the environment over time. Designers in class frequently express temporal requirements in their problem formulations; roughly 2% of all concepts coded to date are explicitly temporal.
Polytely	A universal phenomenon in design is to meet simultaneous, contradictory goals. In 2006, one of the techniques instructors in the BID class taught was to characterize a design problem as an optimization of competing constraints (Helms, Vattam & Goel, 2008). One of the fundamental concepts in the design tool TRIZ (Altshuller, 1984), is the notion of technical contradictions. Many engineering software packages come with design optimization algorithms for finding the optimal values of particular competing variables for a design.

Table 6-12. Design behaviors associated with complex problem solving.

Characteristic	Design Goal	Example Behaviors
Intransparency	Intransparency requires from the problem solver the active acquisition of information.	Knowledge seeking behavior is deeply engrained in the BID class culture, starting with lectures on information search strategies, and requiring students to provide credible sources/papers supporting research on both problem definitions and biological analogues.
Complexity	Complexity demands from the problem solver a simplification through reduction.	Problem decomposition is a well-documented design strategy for reduction, seen often in biologically inspired design (Yen et al, 2011, 2014). Problem partitioning by domain, such as electronic systems versus mechanical systems, is also common.
Connectivity	For making mutual dependencies understandable, a model of the connectivity is required from the problem solver.	Modeling problems and solutions is embedded in classroom behavior in a number of ways, from the use of SBF and SR.BID models, to emphasis on mechanistic explanation (causal models), to quantified analysis of key aspects of the designs model which requires detailed articulation of the relationships between the design and quantifiable engineering principles.
Dynamics	Dynamics requires from the problem solver the consideration of the factor “time.”	Emphasis on mechanism and causal reasoning demonstrates the consideration of time, as well emphasis on time-based environmental factors such as day-night cycles, and user activity patterns.
Polytely	Conflicts due to antagonistic goals require the forming of compromises and the definition of priorities.	Requirements gathering documents often couch requirements in terms of “needs” versus “wants”. Student materials assessments provide

6.3.2.4.5 Hypothesis P.2.2: Designer Problem Strategies

Table 6-13 provides a small conjectured set of design problem strategies relative to the first two goals in hypothesis P.2.1. Assuming the problem goal is known, each design strategy is comprised of several components as follows:

(a) One or more *concept targets* that serve as both the index to required knowledge, and the focal point of the transformations.

(b) One or more *possible transformations*, that determine what kind of knowledge is required and how that knowledge is applied to achieve the strategy,

- (c) **Knowledge requirements**, determined by the transformations to be applied
- (d) An initial **problem state**, and
- (e) A resultant **problem state**.

Table 6-14 provides a more detailed description for each of the hypothesis strategies in terms of the five strategy components.

Table 6-13. Hypothesized design problem strategies associated with designer problem goals.

Goal	Strategy	Description
Active acquisition of information	Breadth-first addition	Loosely related concepts are added in a breadth-first fashion, expanding the design space.
	Depth-first addition	Sub-concepts are added to existing concepts, generating conceptual depth for a particular concept.
	Relationship addition	Relationships are established between concepts
Simplification through reduction	Elimination	Concepts are removed from consideration in the problem space
	Decomposition	Concepts are divided into sub-concepts that can be considered independently; an interface may be necessary.
	Abstracting	One or more concepts of the same type at the same level of abstraction give rise to a concept at a higher level of abstraction.

The **concept target**, describes the general characteristics of the initial concept that the design seeks. The criteria for the selection of a specific concept target depend on the perceived salience of that concept to the particular designer problem goal. For example, if the problem is intransparency, then it may be a combination of lack of detailed understanding about a concept, and the saliency of that concept to current designer thinking that focuses the designer on that concept. Meta-knowledge may also play a role; that is, knowing that the knowledge requirement for a concept may be easily fulfilled, for

example by a design team member or an article at-hand, may make focusing on one concept a better choice than on another for which the ability to fulfill the knowledge requirement is unknown, or is known to be difficult

Table 6-14. Design problem strategy details.

Strategy	Concept Target	Transformations	Knowledge Requirement	Initial State	Resultant State
Breadth-first addition	An initial concept is selected to use as an index for adding related concepts.	Refining, associating, or disconnected addition.	Concepts related to the selected concept.	C_1	$C_1 \rightarrow C_2$ C_1, C_2
Depth-first addition	An initial concept is selected, for which the designer wants more detailed knowledge.	Shifting down, or shifting down + associating, or decomposing	A more detailed account of the selected concept.	C_1	$C_1 \rightarrow C_{1-1}$ $C_1 \rightarrow C_{1-1}, C_{1-2} \dots C_{1-n}$
Relationship addition	An initial concept is selected to use as an index for adding relationships.	Connecting, or switching connection	Relationships among concepts.	C_1, C_2 $C_1 \rightarrow C_2, C_3$	$C_1 \rightarrow C_2$ $C_1 \rightarrow C_3, C_2$
Elimination	An initial concept is selected to use as an index for adding relationships.	Dropping, or suppressing	No additional knowledge is required.	C_1, C_2	C_1
Decomposition	An initial concept is selected which is to be used to sub-divide the design problem	Decomposing	A more detailed account of the selected concept.	C_1	$C_1 \rightarrow C_{1-1}, C_{1-2} \dots C_{1-n}$
Abstraction	One or more concepts are selected which are to be used to generate an abstraction.	Shifting up, or abstracting	Commonalities among concepts.	C_{1-1} $C_{1-1}, C_{1-2} \dots C_{1-n}$	$C_1 \rightarrow C_{1-1}$ $C_1 \rightarrow C_{1-1}, C_{1-2} \dots C_{1-n}$

The knowledge requirement assumes that a designer has or can acquire the knowledge in question to execute the transformation. For example, if executing a decomposition transformation on a function (say “clean self”), the knowledge of the sub-functions for at least one method of “clean-self” must be known (for example, “lather”, “rinse”, “repeat”). The knowledge requirement may be a primary driver in selecting the strategy and concept target. For example, given the choice between “dissipate heat” and “clean self”, a chemical engineer with expertise in detergents may focus on a “clean self” function for which they feel they already know a great deal of related concepts. An engineer with deep thermodynamics background may focus on “dissipate heat”.

6.3.2.4.6 Method P.2.2

Generating a complete taxonomy and supporting evidence for designer problem strategies is a future research topic. It may be possible to provide indirect evidence for the existence of DPSs using existing data in two ways. First, I may infer DPSs through differential analysis of SR.BID encoded designer problem models. From the differential analysis, a set of transformations used is inferred. Patterns of transformation that match those specified by the DPS, are taken as representative of the underlying DPS used to accomplish that goal. A narrative is constructed supported by available documentation by which a designer could reasonably be said to have execute that DPS. Second, I may infer the execution of a DPS directly from (un-coded) design problem documents provided by designers as part of in-class assignments. These documents provide “resultant-state” snap-shots of design problems resultant-states of problems only, in this case implying both the initial state of the problem and the DPS that could have been used to arrive at the observable resultant state. Ideally, multiple example of each kind of analysis.

Future evidence of design problem strategies may be gathered through new studies using a combination of talk-aloud protocols and SR.BID differential analysis of design episodes.

6.3.2.4.7 Hypothesis P.2.4: Transformations

Each strategy is based on the application of one or more transformations. Based on my own observations of problem formulations in the context of BID and assuming the SR.BID content account, I hypothesize the following set of transformation primitives over the content of problem formulations in BID.

Table 6-15. The hypothesized set of problem transformation primitives.

Type	Sub-Type	Tertiary Type	Start State	End State
Addition	Refining		A	$A \rightarrow (r)B$
	Associating		A	$A \rightarrow (a)B$
	Abstraction shifting	Shifting-up (zooming out)	A(1-1)	$A(1) \rightarrow A(1-1)$
		Shifting-down (zooming in)	A(1)	$A(1) \rightarrow A(1-1)$
	Induced abstraction		A(1-1), A(1-2)	$A(1) \rightarrow (A(1-1), A(1-2))$
	Decomposing	Conjunctive	A(1)	$A(1) \rightarrow (A(1-1) \text{ AND } A(1-2))$
		Disjunctive	A(1)	$A(1) \rightarrow (A(1-1) \text{ OR } A(1-2))$
	Disconnected		--	A

Table 6-15 continued

Removal	Suppressing		$A \rightarrow B$	$A, \neg B$
	Dropping	Disconnected	A	$\neg A$
		Dependent chain	$A \rightarrow B$	$\neg A, \neg B$
		Partition	$(A \rightarrow B) \rightarrow C$	$\neg(A \rightarrow B), C$
	Reemerging	Related	$A \rightarrow B: \neg B$	$A \rightarrow B$
		Novel	$A \rightarrow B: \neg B$	$C \rightarrow B$
Connecting	Connect		A, B	$A \rightarrow B$
	Disconnect		$A \rightarrow B$	A, B
	Switch Connection		$A \rightarrow B$	$A \rightarrow C$
Organizing	Partitioning		A, B, C, D, E	$(A \rightarrow B) \rightarrow C \rightarrow (D \rightarrow E)$
	Decoupling		A, B, D, E	$(A \rightarrow B)(D \rightarrow E)$
<i>Changing</i>	Swapping		$A \rightarrow B \rightarrow C$	$A \rightarrow D \rightarrow C$
	Incrementing		$A \rightarrow B \rightarrow C$	$A \rightarrow B' \rightarrow C$
	Composition		A, B	$(A+B)$

A complete description of each of these transformations with examples is available in Appendix H.

6.3.2.4.8 Method P.2.4

My goal is to establish the set of transformations that can occur to a problem model during problem evolution. One method to validate the existence of at least a partial set of

the transformations is to trace the complete trajectory of a small model sub-set (one concept, the function “clean-self” and all concepts related to it) through the entire design trajectory. For this model sub-set, I map the transformations necessary to move from one model to the next to the observed state changes. This provides a step-by-step illustrative example of how transformations may be applied in sequence to create larger shifts in the problem model over time. It also provides supporting evidence for the existence of those transformations that are witnessed in the model sub-set trajectory.

6.3.2.4.9 Evaluation P.2.4

The design trajectories from 2009 derived from SR.BID may be considered valid to the extent that they were validated in study 3.2 (see section 7.2.2). To the extent that a pre- and post- state exact match can be determined in the data, and to the extent that the indicated transformation could be reasonably said to have occurred for the content in question, I will designate such a transformation as being instantiated.

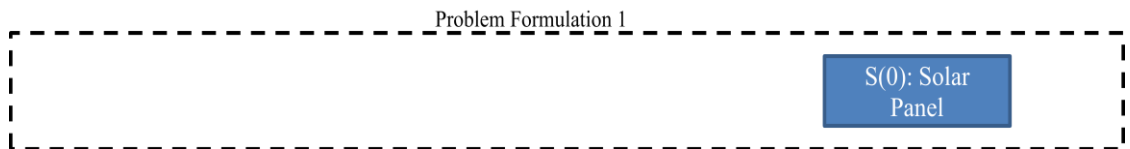
Because the models represent gaps of 3 weeks, it could be the case that a combination of transformations occurred in sequence to arrive at certain observed patterns between models. This creates set of alternative explanations for the observed patterns. Since the purpose of this study is to show that one of the set of transformations may in principle exist, I will make the simplifying assumption that the most straightforward transformation between problem models is the correct interpretation. In cases where the result is still ambiguous, I will provide alternative explanations.

6.3.2.4.10 Results P.2.4

I begin by looking at the function concept “clean (self)” and concepts directly related to it. In some cases in order to understand the transformation used to on a related concept,

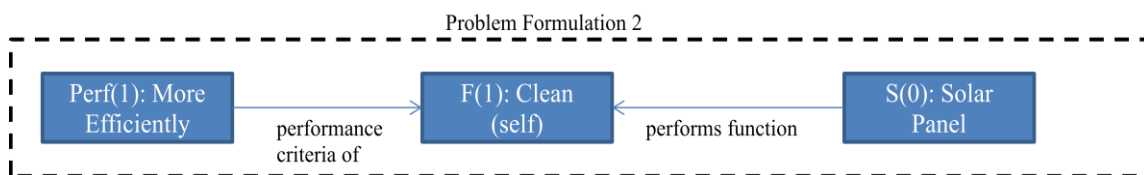
it will be necessary to show concepts two steps removed from “clean (self).” To avoid regressing into an explanation of all related concepts, I will not be discussing the transformations that led the addition of concepts twice removed from “clean-self.” These are shown with dotted line borders and white backgrounds with black font.

Problem formulation one does not contain the function “clean (self)”, although it does contain a solution concept “solar panel”, which will be connected to “clean (self)” in the next iteration. I show “solar panel” here so we can understand the transformation in the next step.



Formulation 1 Transformations			
Step	Concept(s)	Transformation	Related Concepts
(start)	S(0)	(start)	--

In the second problem formulation, the function concept “clean (self)” is added as a function of the solar panel. This is an example of an *associating addition*. Next, we note that the performance criteria concept of “more efficiently” is added as a modifier to the “clean (self)” function, which is an example of a *refining addition*.



Formulation 2 Transformations			
Step	Concept(s)	Transformation	Related Concepts
1	F(1)	Associative Addition	S(0)
2	Perf(1)	Refining Addition	F(1)

In the third problem formulation, the function concept “clean(self)” is much more deeply embedded in the problem structure. Firstly, the concept is related to three different solutions: the original “solar panel” solution; the “lily pad” a biological solution, and a “nano-scale surface solar panel” which is conjectured solution. Furthermore, we see two additional related functions were added, “respond to environment”, and “dissipate heat”, the later of which is related to the “solar panel”. An additional fourth function “make dynamic (self)” is added at one level of abstraction higher. We also note that while the performance criteria “more efficiently” still exists, it has been connected to a different function, “generate energy”.

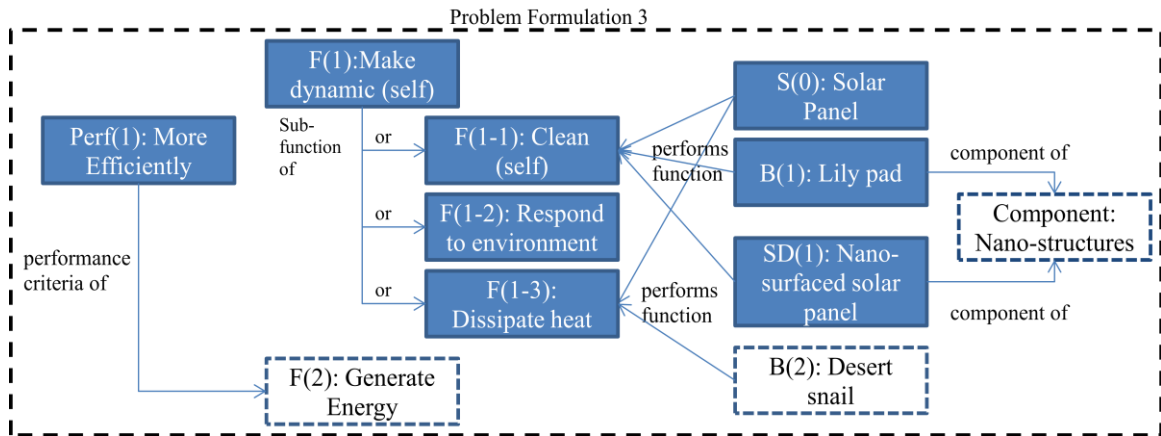
Because these are snap shots and we do not know the precise order in which the concepts were added, there is some ambiguity as to what transformations concepts were added to the problem formulation. I note where I have made assumptions.

First, I assume that the lily pad was added as an associated concept (an *associating addition*) with the “clean (self)” function. The function “dissipate heat” likewise appears to be added as an *associating addition* with the biological source “desert snail”. Let us also assume that the concept “respond to environment” was added as a *disconnected addition*. Following this addition, from the concepts “clean (self)”, “dissipate heat”, and “respond to the environment”, the designer induces the function (an *induced abstraction*) “make dynamic (self)”.

Two addition changes require an explanation. The first is changing the relationship (*connection switching*) of the performance criteria “more efficiently” from modifying “clean self” to “generate energy”. The second is the emergence of the concept of the new solution “nano-surfaced solar panel”. The genesis of this concept appears to be from the

composition of the component specification of the lily pad (nano-structures), with the existing solution solar panel.

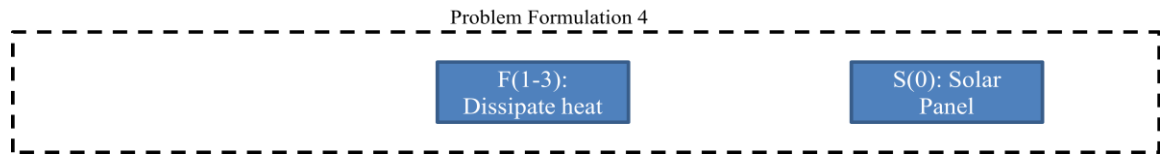
We also note that there appears to be a connected sub-graph involving $f(1-1)$, $b(1)$, $spec(1)$, $sd(1)$, and $s(0)$. Likewise, there appears to be a connected sub-graph involving $s(0)$, $f(1-3)$, and $b(2)$. We call each of these sub-graphs, which have multiple conceptual types, a **partition**.



Formulation 3 Transformations			
Step	Concept(s)	Transformation	Related Concepts
1	B(1)	Associative addition	F(1-1)
2	B(2)	(second order)	--
3	F(1-2)	Associative addition	B(2)
4	F(1-3)	Disconnected addition	--
5	F(1)	Induced abstraction	F(1-1, 1-2, 1-3)
6	F(4)	(second order)	--
7	Perf(1)	Connection switch	F(4)
8	Spec(1)	(second order)	--
9	SD(1)	Composition	S(0), Spec(1), F(1-1)
10	F(1-1), B(1), Spec(1), SD(1), S(0)	Partition (1)	--
11	F(2), B(2), S(0)	Partition (2)	--

Most of the concepts related to “clean self” are removed for the final presentation. The “clean (self)” function, and nearly all concepts in the first partition (except the solar panel itself) are removed. I call this removal of an entire partition a **partition deletion**.

Since perf(1) was a refinement of f(2), when f(2) was deleted perf(1) no longer had a role, and so it was also deleted. I call this a *dependency deletion*. In as much as f(1) was an induced abstraction over f(1-1), f(1-2), and f(1-3), and since two of the three were removed, it appears that f(1) no longer served a purpose, and was also deleted, another *dependency deletion*.



Formulation 4 Transformations			
Step	Concept(s)	Transformation	Related Concepts
1	F(1-1), B(1), Spec(1), SD(1)	Partition deletion	--
2	F(4)	Single deletion	--
3	Perf(1)	Dependency deletion	F(4)
4	F(1-2)	Single deletion	--
5	F(1)	Dependency deletion	F(1-1, 1-2)
6	B(2)	(second order)	--

This study provides a trace of a small portion of a problem model as it changes over time. I use the transformation taxonomy to show how, at the level of operations over concepts, the problem model changes incrementally. This study provides some insight into the potential power of the transformation taxonomy both as a descriptive tool, and prospectively as a tool for use in intelligence systems.

6.3.2.4.11 Hypothesis P.2.5: Problem-solution memory

The final component of the PE.BID theory is a theory of memory content and organization, in particular the partial integration of a problem-solution memory. First, I assume that SR.BID, as a content account of problem formulation, forms the basis for the design problem organization in memory. Next, I note that the core concepts in the

problem formulation: function, environment, performance criteria and specifications/constraints, have corresponding concepts in solution representation. However, it is not necessarily the case that all concepts in a problem representation are also represented in a solution; for example, while a problem may specify a function at a high level, the lower level functions which a particular solution implements may not be. Likewise, not all performance criteria in a problem formulation may be relevant to a particular solution. Moreover, evidence shows that all problems defined in the context of BID, are defined in terms of one or more solutions.

Take for example the problem of moving a skier up a mountain. Concepts associated with a ski-lift will (for anyone who has gone skiing) may significantly overlap with the problem formulation, whereas much fewer concepts associated with, say, an automobile will overlap, and even fewer still with a polar bear. In fact, the overlap of concepts between polar bear and moving a skier up a mountain is so low, it is likely the polar bear solution concept would never be activated without prompting.

The memory hypothesis makes two claims about the initial organization of design memory. The first is that problem formulations and solutions are linked at the level of the concepts described in SR.BID. Solutions and problems are abstractions indexing different arrangements over the same set of core concepts. The second is that the degree to which they are linked varies based on concept type: in particular for novices in BID, they are linked in decreasing order of frequency by function, environment, performance criteria, and specifications/constraints. This may be a result of both the difference in domains (engineering and biology) and the level of experience of the designers.

Table 6-16. Problem-solution model relationships.

Solution Concept	Problem Concept	Relationship
Structure	Constraints/Specifications	The problem model either prescribes specific components and relationships (specifications) for the structure, or a set of parameters which the structures must meet (constraints).
Function	Function	The problem model provides a set of functions the solution concept must perform
Behavior	Performance Criteria	The performance criteria aspect of the model provides a set of criteria against which the behavior of the solution can be measured
External Input	Operational environment	The operational environment aspect of the problem model provides the set of potential external influences which may act on the system.
Structure, Behavior	Benefits/Deficiencies	This aspect of the problem model serves to draw specific attention to aspects of existing solutions, (usually behaviors or structures) that should be included (benefits) or improved upon (deficiencies).

The concepts of SR.BID provides a list for concepts used to define a problem specification. There is always a relationship between problem concepts and solutions concepts. The problem model provides the set of criteria which must be met for a solution to be said to have solved that design problem.

For example, while a solution model is not specified in terms of constraints, the constraint concepts (as defined in SR.BID) are correlated directly with the elements of the structural model. More concretely, one can include a constraint in a problem model that recyclable materials must be used; this is then reflected in the structure model of a solution in terms of the specific material used, for example parts made from recyclable plastic. Table 6-16 shows the relationships between problem and solution models.

This suggests that in the PE.BID process account, that memory for problems and solutions are tightly connected, and possibly (in the case of function, for example) overlapping. In the observed cases of novice designers in biologically inspired design, the degree of integration is stronger in function, and progressively less strong from operating

environment, to performance criteria, to specifications and constraints. This suggests that the degree to which overlap/integration occurs may depend on the concept type. This integration level is probably dependent on at least two other variables: domain of problem and solution, and expertise of the designer in each domain. We would expect to see tighter integration for near domains between problem and solution, and tighter integration for experts versus novices.

6.3.2.4.12 Method P.2.5

I have already documented some aspects of the connection between design problems and existing solutions; for example, from coded problem formulations we see that all problem models cite existing solutions. The degree to which they cite different types of concept varies by conceptual type.

Many future lines of research are open in this direction. First, one may attempt to understand how the connectivity between solutions, both existing and conjectured, changes through a project lifecycle from conceptual design to implementation. Second, one may attempt to understand how the connectivity changes from novice to expert designers, and from the same versus from different domains. I defer additional questions regarding the organization of problem-solution memory to future research.

6.3.3 Summary of Study P.2

In this study, I provided a high-level process model for problem evolution in biologically inspired design. This process model describes **why** problems evolve in terms of designer problem goals, which are a type of general problem solving methods used when confronted with complex problems. In the domain of design, these goals are instantiated in terms of design problem strategies, which are used to direct the

transformation of a problem. These strategies focus on applying one or more low-level transformations to realize the desired problem goal, for instance to reduce complexity.

In order to execute these transformations, the strategy must be aware of where the transformation is to take place, and in many cases the strategy must deploy additional information. How does the strategy access this information? I conjecture that memory is organized such that problem concepts may be used to index a connected problem-solution memory, and it is from this problem-solution memory that the requisite knowledge is pulled. I have shown in previous studies that problems are situated in terms of existing solutions; almost universally in terms of existing man-made solutions. Furthermore, from the detailed study of transformations, I make the following claim.

6.4 Claim 3

The PE.BID transformation set provides a minimum set of primitive transformations for explaining incremental problem state transformations.

7 THE FOUR-BOX METHOD OF PROBLEM FORMULATION

In this section, I will review the four-box method of problem formulation, a tool that addresses the challenges of problem definition in the BID classroom. In the *in situ* studies conducted in 2006 and 2007 I found that students experienced considerable difficulty in formulating design problems of their own creation. Since design problems provide both index and evaluation criteria for the biological analogies, poorly defined problems yield additional challenges, including difficulty searching for and evaluating analogies. While process and tool support were provided to assist student designers with the task of search, problem formulation and analogy evaluation remained unaddressed. To address this challenge I implemented the four-box method of problem formulation, which is based on the SR.BID content account. The four-box method of problem formulation was extended to analogical evaluation through a tool called the T-chart method of analogical evaluation. Based on the success of these tools, SR.BID was tested as an underlying framework for distributed knowledge acquisition for biologically inspired design, through a web-based application.

I will begin this section with a recapitulation of the relevant findings from the *in situ* classroom studies. I will then define the four-box method of problem definition and its intent, explain how it was implemented in the classroom environment, and provide results from implementations conducted in 2011 and 2012. I will then discuss the extension of the four-box as means for analogical evaluation, using a tool called the T-chart method of analogical evaluation. I will conclude with a discussion of the implementation of the SR.BID Web Application, a web-based version of the tool implemented in the second half of the 2012 classroom.

7.1 Problem Formulation Challenges

In the observational study documented in section 4.1.3.3, I enumerated a number of common errors that student design teams make.

7.1.1 Error 1. Vaguely defined problems.

Problems that are nebulously defined, such as “lowering our dependence on oil,” or “protecting a cell phone,” are either too vague to yield to functional descriptions (resulting in no found analogies), or result in too large a search space (resulting in too many poorly matching analogies). This observation suggests that *student designers have difficulty formulating design problems at the correct level of abstraction.*

7.1.2 Error 2. Poor problem-solution pairing.

Frequently, designers match problems to biological solutions based on vague or superficial similarity, such as matching “making a better dishwashing detergent” with the “cleaning properties of the lotus leaf.” While the function “cleaning” is similar, the lotus leaf relies on the structural details of the structure to be cleaned, which a detergent cannot manipulate. This observation suggests *that shallow understanding of the problem, the biological source solution, or both, may lead poor analogy choices.*

7.1.3 Error 3. Misapplied analogy.

When making an analogy, superficial or high-level matches are often forced into an incongruent solution space, yielding flawed solutions. For instance, a two-way traffic optimization algorithm derived from ant foraging behavior, applied directly to a throughput traffic optimization problem yielded an erroneous model. Fixation on this

erroneous model resulted in three design revision attempts prior to it being discarded. This observation suggests, that even when a good analogy is selected, *identifying, understanding and adapting to critical differences between the problem and the solution is critical in making an effective transfer.*

These observations supported the following findings.

7.1.4 Finding 1: Designers struggle with design problem formulation

I observed student design teams formulate and evolve (incrementally reformulate) their design problem, often with radical transformations which require discarding or modifying existing solution design(s). This struggle is ongoing, and often dramatic. For example in study 3.1 I show that 80% of functions were discarded along the way and only 2 of 25 initially proposed functions remained through to the end of the design.

7.1.5 Finding 3: Designers have difficulty making “correct” analogies

I found that the difficulty in defining the design problem translates into certain difficulties in making analogies. This is not unexpected since analogical theories stipulate a “target problem” which forms the basis for many processes of analogy, from retrieval to mapping to transfer to storage. In my initial studies I observed that students both had difficulty (a) finding appropriate analogies, and (b) applying the analogy correctly to their design problem, both of which would result from a poorly defined problem.

These three observations lead to the definition of my third research problem.

7.2 Research Problem 3

Problem formulation in BID plays an important role in searching for and evaluating analogical sources. I observe that students struggle with problem

formulation in BID, and consequently with analogical evaluation. However, there exists no systematic support for problem formulation specific to the BID context.

This research problem arises from a combination of the observations in the previous section, and from the circumstance of the specific study context. In this context formal structures for supporting problem development and for supporting analogical evaluation were not made available to students. As I documented in the review of problem representations afforded by standard engineering tools (see study 5.1), there currently exist *no problem representations designed explicitly for supporting biological analogues, nor for supporting the process of analogy making and evaluation.* Reasoning that SR.BID, developed to encode the problem formulations that do occur in BID, may be used to scaffold the activity of problem formulation in design, I developed the four-box method to support the problem formulation activity.

7.3 Study I.1: Accuracy of Student Use of the Four-Box Method

7.3.1 Study Motivation

The motivation of this study is to understand how SR.BID can be used to improve problem definition activity in the context of the biologically inspired design class. My initial attempts to implement structured representations for solutions using the DANE system based on the SBF ontology met with mixed results. (Vattam et al, 2010). In that implementation, one student found the representations helpful for organizing and reasoning about their designs, one found it beneficial for decomposing problems, and one found the idea of a resource database helpful. Another student reflected that DANE “did not help with communication” and “did not feel that it had great potential as an aide.”

Only one student appeared to engage with the tool beyond the required, allocated class interactions. In our own reflection the implementation team reasoned that several aspects of the implementation influenced participation. First, while the tool was designed for the BID environment, the degree to which DANE was integrated with other course materials was minimal. Second, the DANE study identified that the use of new, structured representations in class depends on the cost to students (of learning and using the representation) versus the perceived, immediate benefit. We calculated that an expert SBF model required between 40 and 100 hours to complete, a high cost to designers, compounded by the pace and workload associated with the BID course.¹⁶ From these results, I conclude that goals of any study of a tool implemented in the context of a classroom, in addition to testing for an experimental effect, should also meet certain ease-of-use and perceived value thresholds.

These ease-of-use and perceived value goals serve an important secondary purpose in a broader context. The developers of BID support tools, including both academic and professional support tools, want to design a scalable solution for providing salient biological knowledge to designers. The number of known and exploitable potential biological systems, sub-systems, and unique functions numbers *at least* in the millions. Any system that hopes to leverage designers (including biologists at the design table) to *generate structured biological and design content* for later use in design must also meet the ease-of-use and perceived value thresholds.

The goal of this study is to demonstrate that the structured representations for biologically inspired design ontology in the form of the four-box method of problem

¹⁶ In end-of-semester anonymous course reviews, workload is the most frequently cited student criticism.

specification, meets the ease-of-use requirements for designers. The next study will address student perceived value. Subsequent hypothesis and studies are suggested at the end of this section for testing a variety of additional effects.

7.3.2 Methodology

The four-box method is implemented within the existing framework of the class as a replacement for generic problem definition assignments at the individual and team levels. Sections 3.3.3.1 and 3.3.3.2 document the problem definition assignments. Assignments are collected and evaluated in terms of number of student assignments completed, and accuracy with which the method is used. Students are provided with a survey at the end of the semester which seeks to understand opportunities for improvement in the four-box method. They are provided with a take-home final reflection assignment, which prompts for open comments about the 4-box method. Conclusions are drawn from student use data, surveys and reflections with respect to the feasibility of the system for systematically encoding problems in biologically inspired design, as well as for design improvements for future implementations.

7.3.2.1 Research Question I.1

To what extent can SR.BID be used accurately for the design task of problem formulation in the context of the BID classroom?

7.3.2.2 Hypothesis I.1

The four-box method (see Figure 7-1) can be used accurately by all students to represent design problems in BID.

7.3.2.3 *Context and Participants I.1*

This study is carried out in the 2011 and 2012 BID@GATech classroom outlined in section 3.3.2. The participants are the student designers documented in section 3.2.3. Students were given the choice to opt-in to these studies at the beginning of class. Participation in this study required no additional effort. Instructors and graders had no knowledge of student participation, and participation had no influence on a student's grade.

7.3.2.4 *Execution of the Study I.1*

In 2008 through 2010, instructors added explicit problem definition assignments to class (see section 3.3.3.1). In 2011 and 2012, I worked with instructors to integrate the four-box method of problem definition with the problem definition assignments. For all problem definition assignments, *except the first assignment and the final report*, instructors required students to use the four-box method of problem definition. In addition to being required for problem-formulation specific assignments, the four-box problem representations were used or referenced in analogy evaluations, materials assessments, and final design presentations and reports. Grading rubrics were changed to reflect the inclusion of the four-box method. In this way, the four-box method of problem formulation was tightly integrated with the overall course. Lecture, assignment and rubric redesign was accomplished by working with instructors for the three months prior to each course. Changes to the course materials were approved by the instructors prior to the beginning of the course.

The first problem-definition assignment (a solution-based problem definition that followed from selecting a biological solution and relevant function) was given in week

two in both 2011 and 2012 and was worded as follows:

Now that you have a solution and a function defined, and you have some insight into how the biological solution works, think about what kind of problem you can solve using it. Ideally, the problem should be small, tractable, and something that you could prototype or implement as a senior design project. Write a succinct one or two paragraph description of the design problem you are trying to solve. You should NOT write about how you intend on solving the problem just yet. Focus on the design problem you are trying to solve and what makes it problematic. There must be good reasons why someone hasn't built a better solution already, right? Consider the existing solutions to your problem and what makes them good or bad. You need to think deeply about the problem you are trying to solve and demonstrate that you understand the problem.

In week seven, I provided students with an instructional lecture for problem definition. This lecture provides an explanation of SR.BID as well as the methodology for using SR.BID called the four-box method, which is used for defining a problem. This method was derived from the major categories developed in SR.BID, and is referred to in assignments alternatively as “structure representations for problems” or the “four-box method.” Figure 7-1 provides the instructional representation of the four-box method, which frames problem description in terms of the four concepts: operational environment, function, specifications, and performance criteria.

Operational Environment	Functions
Specifications	Performance Criteria

Figure 7-1. Four box method of problem definition.

In addition to providing the high-level framework, the lecture provides slides that outline sub-categories associated with each major category, as well as an interactive example of a design problem categorized in the four-box framework. During the interactive exercise, a one paragraph design problem is provided to the students, and they are requested to generate their own four-box description. After 10 minutes, the class generates a collective four-box description. The discussion helps highlight differences in problem interpretation, and clarifies distinctions among categories.

In both 2011 and 2012, an individual problem-definition assignment using the four-box method was given in week seven following the lecture. The assignment consisted of three parts: (1) identifying a large number of common problems, (2) selecting and generating a specification for one of the problems, and (3) decomposing the problem into sub-problems. Only the second part of the exercise was considered for this analysis. The following excerpt from the instructions highlight the differences between the four-box method and the approach used in week 2:

“Now that you have a problem in mind (we’ll call this the main problem), use the Structured Representation for Problems that was covered in lecture to write minimum one-page description of the design problem you

are trying to solve. After you write up your problem, follow the four box process outlined in lecture to define your problem in terms of (1) environment, (2) function, (3) specifications/constraints, and (4) performance criteria and why each is important.”

The assignment emphasizes the four-box (or Structure Representation for Problems) method, again drawing attention to the four key categories: environment, function, specifications/constraints and performance criteria. Technological support was also provided for students using MS Office Excel and Powerpoint templates, systems with which students are already familiar, and are easily integrated into their work process. These technology interventions provide assignment scaffolding, without introducing the additional workload or cognitive effort of learning a new tool or technology. In study 5.3, I investigate the implementation of a customized web-based platform for supporting the four-box method of problem specification.

7.3.2.5 Data I.1

In both 2011 and 2012, assignments were collected electronically and available to researchers for analysis after they were graded. Late assignments and assignments not submitted electronically were not available to the researchers. A total of 31 assignments out of a population of 39 students were collected in 2011, and 33 out of a total population of 34 students were collected in 2012. Students articulated answers to assignments using one of four modes of expression: narrative, list, four-box, or incomplete.

In narrative mode, responses were written in paragraph format, with categorical delineations marked using one of the key terms either as a paragraph heading, or as part of the narrative structure. Half of the 64 assignments used the narrative mode. The following provides an example of a narrative method of expression:

Operational Environment

Smart Phones are located everywhere and are used throughout the entire day. They need to be able to be used in all climates and also withstand pressure and other physical aggravations. There is nothing worse than that feeling of dread when you drop your phone, and hope that it hasn't broken.

Functions

A phone's function is pretty straightforward; it is used as a communication device. People rely on phones to get to where they need to go, connect with people, check their email, and stay up to date on current situations.

Specifications

Most smart phones are made of relatively cheap exterior materials. The main issue when pressure is exerted on a phone, or it is treated in an aggressive way, is the optical glass screen. This screen easily cracks/breaks making the phone unusable until either the screen is replaced, or a new phone is purchased. The cost for the screen is relatively inexpensive, however the repair costs are between \$100 and \$200 for the product to be fixed. The rest of the external materials are relatively cheap as well.

Performance Criteria

The function that the screen for the iPhone must perform is to be touch-sensitive and protect the underlying screen. A crack ruins performance and causes a lot of issues. The glass must be able to register touch while still being strong enough to withstand force.

When expressing the problem using the list mode, the student expresses the problem in terms of a list of concepts, each associated with one of the four-box categories. A total of 8 of the 64 assignments were expressed as lists. The following is an example.

- 1. Environment:*
 - Damp environment*
 - Temperatures from -5 to 90 degrees celcius*
 - Mildly abrasive surfaces*
 - Greasy, animal fats*
 - Lots of soaps and solvents*
 - Air temp maintained at 20-25 degrees*

No direct sunlight
Intermittent periods of saturation

2. Function:

Holds water
Holds soap
Makes lather
Physically removes food from dishes

3. Constraints

Must fit in hand
Must not harm dishes
Must be inexpensive

4. Performance criteria

Target dimensions
Maximum cost
Useful life
Absorbtion
Drying time
Amount of bacteria

When using the four-box method, students divide terms in a four-box chart, as was demonstrated in class. A total of 21 students expressed problems using the four-box mode of expression. Figure 7-2 shows an example of a student using the four-box mode of representation. Student using the four-box mode of representation always use the same format with respect of ordering of the boxes, that is, in order clockwise from the upper left: operational environment, functions, performance criteria, and specifications. There is no intended meaning associated with the positions of these boxes.

Operational Environment	Functions
<ul style="list-style-type: none"> • Location: public restroom • Time frame: seconds • Possibly high traffic settings • Exposure to soap, water, bacteria • Must not obstruct entrance 	<ul style="list-style-type: none"> • Detect contact • Remove materials/sanitize • Detect sanitization threshold and disengage response • Retain normal function as door handle
Specifications	Performance Criteria
<ul style="list-style-type: none"> • Cost (especially relative to current handles) • Ease of installation • Material must have comparable structural integrity to current handles (i.e. not malleable) • Must not be in anyway hazardous to users 	<ul style="list-style-type: none"> • Bacterial/material density after sanitization • Response time • Lifetime (cycles) • To what extent is the solution noticeable to the customer-seamlessness

Figure 7-2. Student sample of four-box design problem description.

Three students that turned in assignments completed the initial part of the assignment, but did not attempt the four-box portion of the assignment. It is unknown why this portion was not completed. I categorized these as incomplete, and dropped these three assignments from the analysis.

7.3.2.6 *Evaluation I.1*

My goal is to evaluate the frequency with which students used the four-box method to complete their assignment, and the accuracy with which students were able to assign concepts to the proper categories. I argue that students' ability to complete assignments

and use the tool accurately provides a good proxy the ease-of use of the tool. Frequency of use is measured as the number of students (or teams) that completed the assignment using the four-box method relative to the number of students that completed the assignment. Accuracy is measured as the ability of those individuals that used the assignment to accurately assign each concept to an appropriate category. To measure frequency, it turned out that there was only one of two conditions present for each collected assignment. Each student assignment either completed the relevant section using the four-box method, or did not complete section at all. Thus the frequency is the ratio of those that completed relevant section to the total number of assignments collected.

In order to measure accuracy, assignments are encoded using the rubric developed in section 5.4. The encoding is straightforward in the cases of list and 4-box representations, where students have already parsed the data into conceptual units. Each concept is assigned a category code using the rubric. Paragraph mode representations closely resemble the data found in section 5.4.3 and 5.4.4, except they are organized into sections that reflect the four-box representation schema. In paragraph mode, sentences are parsed into conceptual units, which are then assigned a concept code. The concept is also assigned a code based on the section in which it was placed by the student (i.e. either operational environment, function, constraint/specification, or performance criteria). The rater-assigned code concept is compared to the student-assigned code, and is evaluated in as either “agrees” or “disagrees”. Because only concept (and not relationship) encoding is performed which I found in previous studies to be more reliable (see section 5.4.3), a single coder was used to encode all of the data. The degree to which, for any category the

two code agree may be expressed as a percentage of total concepts in agreement over total concepts encoded.

Accuracy is compared between-groups for differences among: gender, major, year (2011 or 2012), and method of student encoding (i.e. paragraph, list, or four-box). Accuracy differences are also compared among the four conceptual types. Additional qualitative data from student perceptions of the four-box method are captured through survey instruments and through an end-of term reflective final assignment, which will be discussed in the next section.

7.3.3 Results of the Study

The frequency of use of the encoding to complete the assignment shows that of the 64 assignments collected, 61 of the students completed the relevant section. Three students did not complete the section, but completed other sections of the assignment. These three students were all in the 2011 class, suggesting a potential between-classes difference in the assignment context. The most significant change between 2011 and 2012 was the inclusion of assignment rubrics with assignments. As a result of this change, students were more aware of the point allocation for each of the sections of the assignment, which itself may have motivated completion. Rubrics also established the criteria against which each section would be assessed, possibly providing the students with more direction in completing that aspect of the assignment. Nevertheless, completion rate on the 4-box portion of the assignment was greater than 95%, providing evidence in favor of students' ability to complete the task.

The overall accuracy of student use of the four-box method, excluding the three students that did not complete the assignment, was measured over a total 1058 concepts.

For all concepts across both years and all concept types, average student accuracy was 85.6% with a min of 52.6% accuracy, max of 100% accuracy and standard deviation of 10.8%. Figure 7-3 shows a histogram of the distribution of accuracy. Only 5 instances were less than 70% accurate, while 46 were greater than 80%. This shows that student designers with less than 90 minutes of instruction are able to create problem descriptions using the four-box method with greater than three-quarters of the students performing with greater than 80% accuracy on their first attempt. Additional analysis shows a more complex set of interactions when more variables are considered.

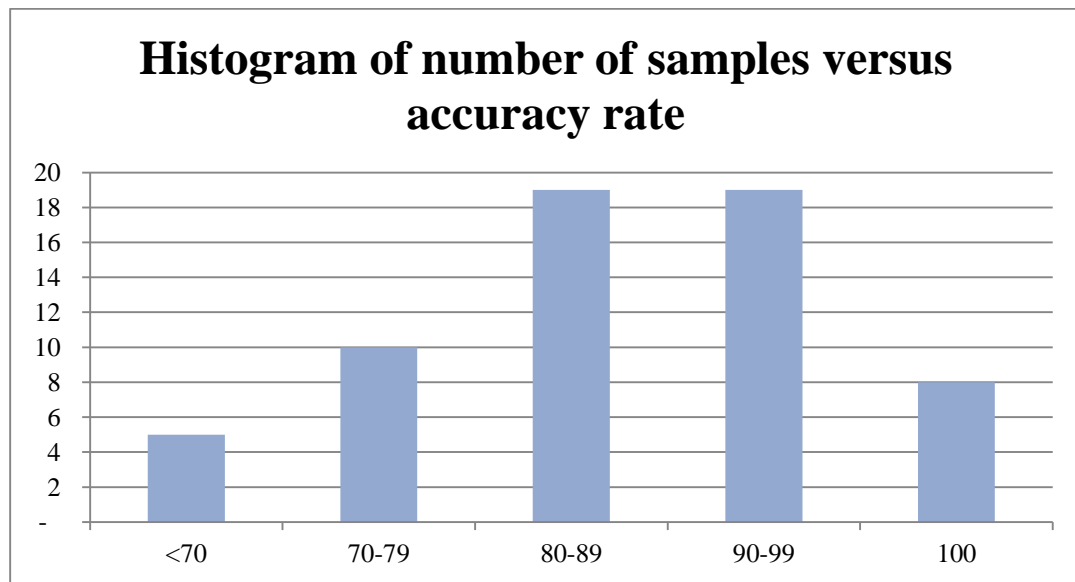


Figure 7-3. Histogram of accuracy of use of four-box method.

Difference in accuracy was measured between conceptual categories, between years 2011 and 2012. Figure 7-4 shows the difference in accuracy among the four basic categories, by year. In 2011 the difference were more pronounced, almost a 30 point difference in the accuracy between students' ability to correctly classify specifications and constraints versus operational environment, with function and performance criteria

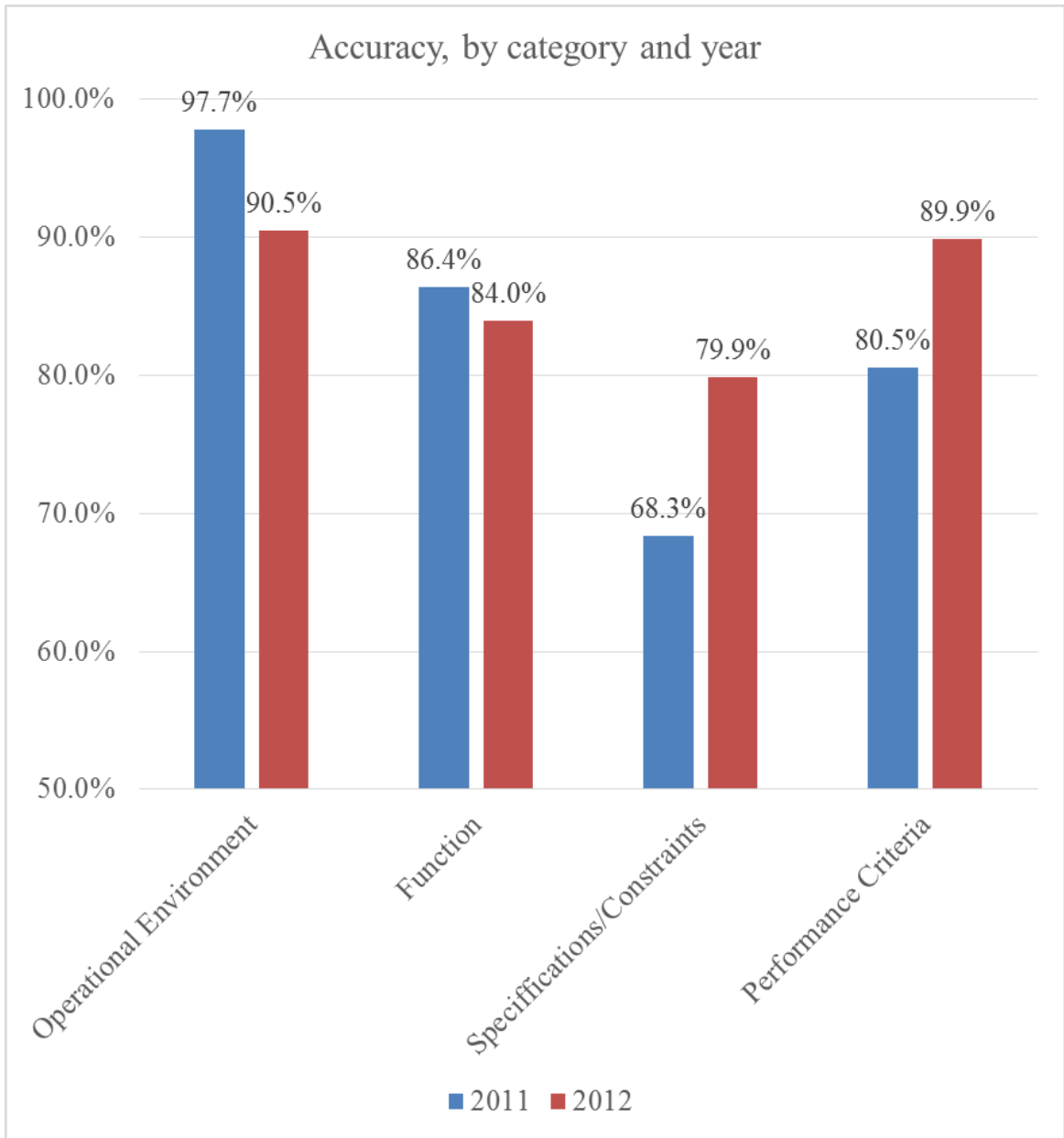


Figure 7-4 Differences in accuracy of 4-box method, by category, by year.

splitting the difference. In 2012 the difference was reduced to almost 10%, but at the cost of a reduction in accuracy in function by almost 7%. This narrowing of the range may be an effect of significantly more concepts present in the 2012 data set, 644 in 2012 versus 414 in 2011, which may have a normalizing effect on the data. As before, changes to the

instruction, the example exercise, or rubric may have likewise had an impact.

Differences in the accuracy among majors were studied. Figure 7-5 shows accuracy by major, grouped by biologists, biomedical engineers, industrial engineers (including architects and computer scientists), and mechanical engineers (including material scientists and polymer and textile engineers). The difference in ranges are smaller, with biologist accuracy slightly lower than others majors.

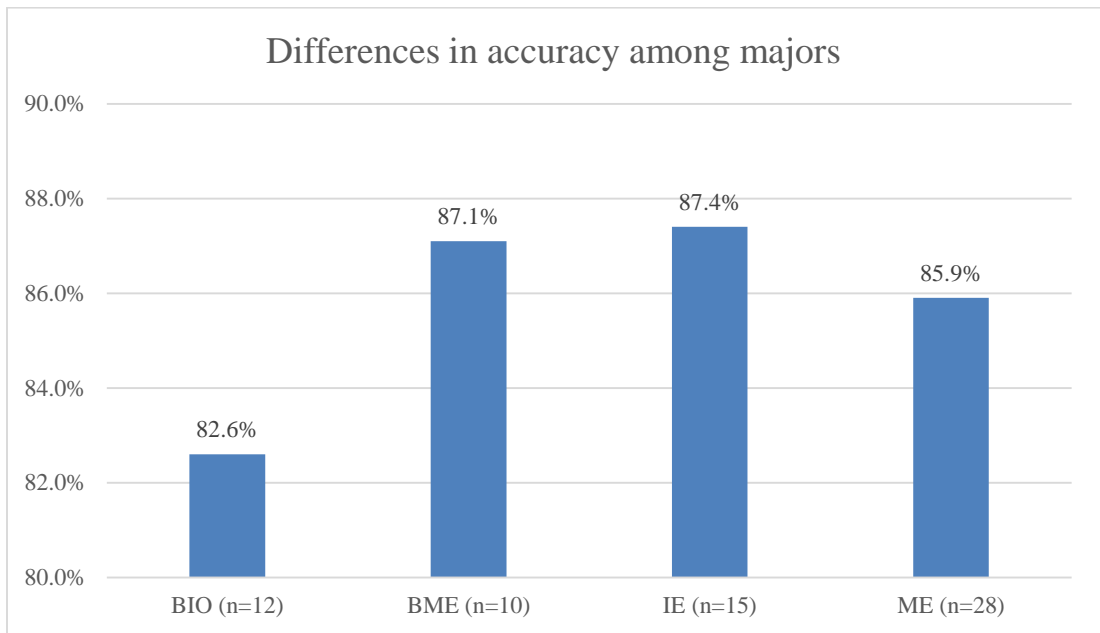


Figure 7-5. Difference in accuracy of 4-box method, by major.

Differences in accuracy based on response mode were also considered. Figure 7-6 shows accuracy by paragraph, list, or four-box response mode. The four-box method shows slightly better performance than list or paragraph modes.

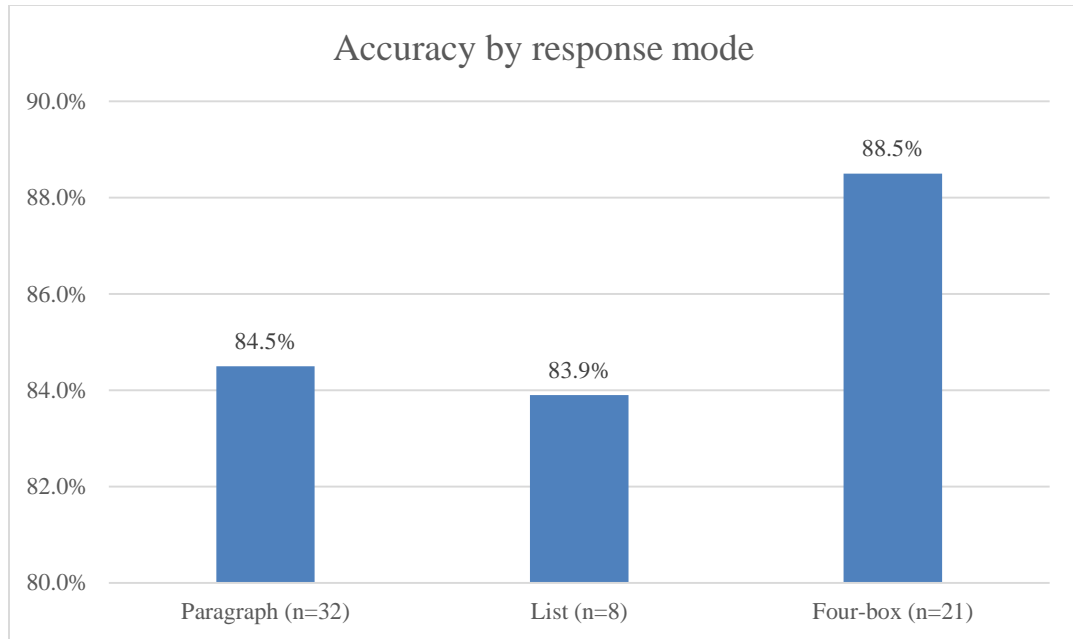


Figure 7-6. Accuracy of use of four-box method, by mode of representation.

There was a very small difference in the mean between males (n=31) and females (n=30) of 0.1%, suggesting no significant variation based on gender.

While each of the individual variable may have an effect on the overall result, it is not clear if there is a significant difference when they are considered collectively. That is, while it appears that mode of response, major and year (2011 vs 2012) have some effect on accuracy, it is not clear whether the effects are significant when observed collectively – that is, it could be that one year had a higher number of biologists using paragraph format, and the combination of these factors are making each variable appear to be more significant than they are. An ANOVA analysis attempts to explain the variance attributable to each variable and for combination of variables. The ANOVA analysis was run using Wolfram Mathematica 9 (version 9.01), Student Addition, testing the effects of year (2011/2012), major (BME, ME, ISYE, BIO), and mode (four-box, narrative),

individually, collectively, and with interactions: year*major, major*mode, and year*mode. The sample size (n=61) was not comprehensive enough to evaluate interactions among all three variables.

Results from single variable tests do not show significance for major (p = 0.714), year (p=.424) or mode (p=.326). Nor do results from combinations of major-mode, major-year, or mode-year, as shown in the following three tables.

Table 7-1. ANOVA analysis of accuracy using major and mode.

ANOVA (MAJOR, MODE, MAJOR*MODE)					
VAR	DF	Sum of Sq	MeanSq	F-Ratio	P-Value
MAJOR	3	0.0164	0.0055	0.4411	0.7245
MODE	1	0.1287	0.0129	1.0385	0.3128
MAJOR*MODE	3	0.0144	0.0048	0.3875	0.7624
ERROR	53	0.6568	0.0124		
TOTAL	60	0.7006			

Table 7-2. ANOVA analysis of accuracy using mode and year.

ANOVA (MODE, YEAR, MODE*YEAR)					
VAR	DF	Sum of Sq	MeanSq	F-Ratio	P-Value
MODE	1	0.0114	0.0114	0.9917	0.3235
YEAR	1	0.0072	0.0072	0.6272	0.4317
MODE*YEAR	1	0.0254	0.0245	2.2094	0.1426
ERROR	57	0.6568	0.0115		
TOTAL	60	0.7006			

Table 7-3. ANOVA analysis of accuracy using year and major.

ANOVA (YEAR, MAJOR, MAJOR*YEAR)					
VAR	DF	Sum of Sq	MeanSq	F-Ratio	P-Value
YEAR	1	0.0076	0.0076	0.6217	0.4339
MAJOR	3	0.0139	0.0046	0.3799	0.7679
YEAR*MAJOR	3	0.0313	0.0104	0.8557	0.4698
ERROR	53	0.6477	0.0122		
TOTAL	60	0.7006			

After executing the ANOVA analysis, the P-values provides a relative confidence rating for the explanatory power of the variable or variable combination with respect to the observed error. In this case, none of the P-values registers as significant. This supports that claim that MAJOR, YEAR and MODE, individually or in combination do not provide significant explanatory power with respect to the observed observation. This is not the same as saying they are insignificant, only that we cannot conclude with any certainty that they have an effect on the result in this context from the available sample. Furthermore, the combination of mode and year shown in table 7-2, while not statistically significant appears to be an outlier. Further investigation shows that the combination of using the four-box mode in year 2011 produced accuracy results on average 4.7% to 6.4% more accurate than results for the narrative mode in either year and for four-box mode in year 2012. It is unknown why this might have occurred.

7.3.4 Summary of the Study

In this study I examined the ability of students to accurately use the four-box method to provide problem descriptions. With less than one class period of training, 95% of students submitting assignments were able to complete the section of the assignment

which required the four-box method of problem description. Of these, greater than 75% of students showed the ability to properly categorize problem concepts according to the four-box scheme, achieving levels greater than 80% for proper categorization of concepts. While some variation existed between year of instruction, major, and modes of reporting, ANOVA analysis did not indicate a statistically significant source of variation from any one variable or combination of student related variables.

The strongest differences occurred between concept types, of which operational environment factors were most accurately classified in each year and constraints/specifications least so. Considering the data used consisted of initial problem formulations, one hypothesis for follow up is that in subsequent problem formulations, as the solution conceptualization takes place, emphasis may shift away from operational environment and focus more on the constraints and specifications placed on the design artifact. As the design product becomes more tangible to the designers, the accuracy of the categorization of the most artifact-centric factors may also increase. Alternatively, instruction and examples should be investigated to ensure this category is receiving complete treatment.

I conclude that across multiple variables, including gender, major, mode of response and year of instruction, students designers in the context of study are quickly capable of producing accurate models of problem formulations using the four-box method. This suggests a low investment cost for student designers for learning and using the tool, one of the success factors identified from earlier studies. This study also suggests that an opportunity for improvement exists in the category of constraints and specifications, which requires additional investigation in terms of the use of that category over time and

in its treatment during training.

7.4 Study I.2. Qualitative Analysis of the Four-Box Method

While the previous study answered whether the four-box method could be used by the student population, it did not address the question of perceived value to the students. In order to answer this question, and to gain additional insight on how the tool might be improved, I used two separate instruments. The first was a short survey conducted at the beginning of the last class in 2012. The second was an analysis of the students' final homework assignment, which asked students to reflect on a number of aspects of the class, including tools and representations such as the four-box method and SR.BID.

7.4.1 Study Motivation

From our previous experience with DANE we know that a student or designer will be more likely to use a tool if they recognize the value with respect to the task at hand. This study provides a clearer understanding of students' perceived value of the SR.BID and four-box support tools after they are used in the applied context.

7.4.2 Methodology

Students are instructed in the SR.BID and the four-box method, as outlined in Study 5.1, and are asked to use the representations and methods in a variety of assignments, including problem definition assignments, materials analysis, and design reports. Use of the methods is required, except for the last design report, and is integrated into the grading rubrics for the assignments, which are available to students. Through the assignments and rubrics, students are aware how and to what degree they are expected to use the tools. On the final day of class, students are given a short survey to identify the

utility students perceived from the use of the SR.BID and four-box method for problem definition. Survey participation is optional, anonymous, and does not affect the student grade. The survey includes rank-order questions, open-ended response questions.

A second instrument, which already exists as part of the class protocol is used to gather additional data. Students are asked a series of questions as a take-home final, and are expected to provide thoughtful, candid responses. Though students need not attend to SR.BID or the four-box method in their answers, many do providing additional insight into student perceptions of the tools.

7.4.2.1 Research question I.2

How are the SR.BID and four-box representations used in the context of biologically inspired design with respect to the task of problem definition?

7.4.2.2 Hypothesis I.2

SR.BID will aid students in defining and communicating design problems and biological systems, and help in understanding and communicating their analogies.

7.4.2.3 Context and Participants I.2

This study is carried out in the 2012 BID@GATech classroom outlined in section 3.3.2. The participants are the student designers documented in section 3.2.3. Students were given the choice to opt-in to these studies at the beginning of class. Participation in this study required no additional effort, with the exception of surveys, which required fifteen minutes at the beginning of one class. Instructors and graders had no knowledge of student participation, and participation had no influence on a student's grade. In the case of surveys, instructors were asked to leave the classroom until surveys were complete.

7.4.2.4 Execution of the Study I.2

Study 4.1 provides detailed explanation of the roll out and use of the SR.BID and four-box support tools in the context of the 2012 BID@GATech classroom. In addition to the problem definition assignments due in week 7 described in study 4.1, SR.BID and four-box representations were also used in the week 9 design reports and presentations. For example in the solution-description section of the design report, students are instructed to “provide an explanation for how the system works, using the structured representation for BID format,” and in the problem-description section they are instructed to “use a four-box description of the problem including function, environment, specification, and performance criteria. SR.BID and the four-box method are also implicitly included in the T-chart method for analogical evaluation (see section 4.3.3.3) which is also required in the second design report.

In week 11, SR.BID and the four-box method are used to frame the materials analysis, as shown from the following excerpt from STEP 1 in the materials analysis homework exercise.

*“STEP 1: Using 4 boxes:
--identify functions. What are the constraints, limitation and abilities of the existing functional materials? How are the new properties of the material going to be achieved? Link functions to specifications.
--identify material properties/constraints/cost (specifications of quantitative/measurable properties: Mechanical electrical, Chemical; Structural [size, geometry, architecture]; cost constraints)
--identify environment (temperature, chemical (e.g. corrosive seawater), electromagnetic [e.g. exposed to uv, ir].
--identify performance characteristics (criteria box)”*

In week 14, for their final design reports, students are no longer explicitly required use SR.BID of the four-box method to describe problems and solutions, to evaluate

analogies, and to assess and select materials. While no longer required, the instruction to use these methods is implicit, and students continue to use them. By the time students are asked to reflect over their experience with the tools, in addition to the problem definition assignment in study 5.1, they have also used the methods at least twice in design reports (design report 2, and the final design) to (a) define their design problem, (b) describe biological solutions, (c) evaluate analogies, and (d) frame their materials assessment.

In the final day of class, students in attendance are provided with a survey. Instructors and graders are asked to step out of class until the surveys are complete. Students are given 15 minutes to complete the survey. A copy of the survey as it was presented to the students can be found in Appendix I. The survey was designed to understand what challenges students faced in design, how well they thought the aspects of the four-box method corresponded with important aspects of problem definition, and in what ways they felt SR.BID and the four-box method assisted them in class. The survey was reviewed by instructors prior to class, and was significantly shortened from the original version to meet the instructor constraints; in particular, to meet time constraints and to maintain high levels of student engagement for the final class. Previous, unrelated experiments with surveys issued at the start of class had a damping effect on student engagement for the remainder of the class.

The survey first gathered name, major/minor, and year in school information. It then asked the following questions:

1. Thinking about your final design problem, what do you think were the most challenging aspects of realizing the conceptual design?
2. Rank the following skills from one to five in order of importance in biologically inspired design. Give a Rank of 1 to the most important skill, a Rank of 2 to the second most important skill, etc. Use each Rank only once:

- _____ Finding relevant biological sources of inspiration
- _____ Understanding the underlying mechanisms of biological sources of inspiration
- _____ Making correct analogies between biological sources of inspiration and design problems
- _____ Defining design problems sufficiently and correctly
- _____ Applying mechanisms from biology to the design problem correctly

What other skills do you think are critical for successful biologically inspired design?

3. On a scale of one to five, where 5 is VERY IMPORTANT, and 1 is NOT AT ALL IMPORTANT how important are the following aspects of a design problem.
- _____ Operational environment
 - _____ Material constraints
 - _____ Manufacturing constraints
 - _____ Function
 - _____ Performance criteria
 - _____ Cost

What other aspects do you think are important for defining a problem in biologically inspired design?

4. On a scale of one to five, where 5 is VERY HELPFUL, and 1 is NOT AT ALL HELPFUL how well did SR.BID help you:
- _____ Define your problem in a constructive way
 - _____ Communicate about your problem to others
 - _____ Define biological solutions in a constructive way
 - _____ Communicate about biological solutions to others
 - _____ Understand the accuracy of your analogy
 - _____ Communicate why your analogy was good/bad

Is there anything you would change about the use of SR.BID in class?

Surveys were collected at the end of the 15 minute period, and instructors were invited to return to the class. Approximately 5 minutes into the survey, some students realized the rank-order instructions changed between questions 2 and 3. The instructions were read aloud to the class, and the differences between the rank order schemes were

highlighted to minimize confusion. Information collected from surveys was not shared with instructors, except in summary/anonymous form, and only after final grades were resolved. Answers to rank-order questions were tabulated in a spreadsheet, associated with the students major and year in school, as provided for on the survey sheet. Answers to open ended questions were transcribed into a spreadsheet. All individual identifying information was then stripped from the data.

In addition to surveys, the class was asked to complete a take-home final, in which they were asked to “reflect upon certain course elements they experienced and processes they learned this semester.” Instructions also asked students to “be thoughtful and reflect on what they learned in class,” and to “take as much time as you need...but be succinct in your answer.” The exam consisted of four graded questions, and two ungraded questions. Final exams were distributed on December 4, and collected electronically, due no later than December 11.

The two graded questions from the take home final exam that were relevant to this study were as follows:

Describe one example of success and one example of failure to apply the techniques learned in class (e.g. SR. BID for solutions, SR. BID for problems, and functional or problem decomposition) to organize knowledge in a manner that is useful for creating/framing a BID project. In your answer please describe one benefit and one challenge associated with the way new design techniques are being taught in this class.

Describe one example of success and one example of failure to apply the processes learned in class (e.g. analogical reasoning [T chart], solution based design) to integrate knowledge in a manner that is useful for creating/framing a BID project. In your answer please describe one benefit and one challenge associated with the way BID processes are being taught in this class.

While final exams were graded by instructors, grades were not factored into the research. As with the survey data, answers to the take home final were transcribed into a spreadsheet. All identifying information was then removed from the answers.

7.4.2.5 Data I.2

In total 23 out of 34 students completed the *survey*. Table 7-1 provides a breakdown by self-reported major of the participants.

Table 7-4. Count of self-reported majors, 2012 student surveys.

Major	Count
BIO	3
BME	7
ISYE	6
ME/PTFE	7
Total	23

All students completed the first open-ended question and the rank-order questions. As the survey progressed, fewer students answered the open-ended portion of questions. Only 20 answered the second open-ended question, 16 answered the third, and 14 answered the final open-ended question. Open-ended questions were answered in bullet-point or short phrase format. Some students provided multiple bullet points or phrases (up to 6) in response to a single question. Table 7-2 provides a sample of 10 answers for the first question: “Thinking about your final design problem, what do you think were the most challenging aspects of realizing the conceptual design?”

Table 7-5. Sample answers to first survey question.

Sample Answers
Being able to move away from sunk cost
Bio-inspired materials are hard to assess for marketable/mass-produced product
Common sense check showed complete redesign required
Feasibility and interaction analysis
Find the biological inspiration
Focusing on biological solutions vs technological and quantification
Justifying principle selected
Manufacturing feasibility
Material selection
Materials - developing a composite

Table 7-3 shows how many bullets or phrases were provided for each of the open ended questions. The first two questions received roughly twice the amount of input as the last two. This may be a reflection on the question, or the order in which the questions were asked, which was the same for each survey.

Table 7-6 Number of phrases or points for each question.

Question	Number of bullet points or phrases
1	42
2	48
3	24
4	22

All 34 students submitted a final exam, and all students that submitted a final exam answered all four graded questions. The reflections were open, such that a student need

not discuss the four-box method or SR.BID representations, although most of them did. Of the 34 students, for the second exam question 28 reflected, at least to some degree, on the four-box method or SR.BID, while 32 did so for the third exam question. All text relevant to the four-box method and SR.BID were extracted from the student answers.

The following are quotes relevant to the 4-box method from answers to the first question:

“When seeing the 4-box method for the first time and trying to recreate it on my own I was lost. My first few attempts at the 4-box method was pitiful and I really think it is a useful technique.”

“But again they [my teammates] taught me how to do it properly and work through the first few with me, and by the end of the course I was using this 4-box technique for almost everything. It helps organize data and information much better than any other technique I had previously used.”

“As for the four-box method for SR.BID, this allows an individual to understand in a deeper way why something acts the way it does. Understanding what something must accomplish, where why and to what extent almost completely describes any function I can think of. This four box method in conjunction with the problem decomposition does a very good job at finding something that can be used in a different way.”

“Teaching us new ways to evaluate and breakdown large problems and solutions made this class different than the other design course I have taken. It forced us as students to approach our thinking in a different methodology rather than forcing us to conform to a strict set of deliverables to be graded on. However this did dramatically increase the work load of the class which made the class more challenging to fit in with other classes being taken. That being said, the new techniques definitely were worth the extra work and are tools I plan to keep using over the years as I progress with my career.”

“The most useful part of any of your structures presented to us was simply telling the students to break problems and solutions down in smaller and smaller parts. The second most useful part was having them identify key

components of the operation environment. Past that, the rest of the 4 box method seems redundant or very secondary at best. It should be more of a two box method.”

Likewise, some comments were focused on the SR.BID method of representation.

“SR.BID was very successful in designs 2 and 3. We had 25 organisms and utilizing SR.BID for the solutions provided a much simpler means of comparing the organisms to each other, trying to focus in on specific organisms and specific qualities. It also proved to be a successful technique in developing the analogs between the problem and biological solutions. Since both were written in SR.BID it was just a matter of check where the problem and solution were the same vs. different.”

“Furthermore the SR.BID techniques for problem decomposition allow for a very detailed emphasis of each aspect of the problem, thus narrowing the problem and providing a more clear idea of the essential aspects of the problem that need to be addressed.”

“One failure of how to organize the knowledge is that your structure tends to be too overworked and begins to exclude things. For instance, I remember when we were doing our bio-inspired behavioral design for robots, one part of our problem said that we had to break down the physical mechanism at several orders of scale and discuss the problems that would be encountered during the scaling, and this sort of wording (married quite exclusively to physical mechanisms vs. behavioral ones) made it tricky to appropriately chop apart our design and fit it into these bins shaped for mechanistic design.”

In total, 39 comments were found regarding the four-box method of problem specification, and 50 comments were found regarding the SR.BID representation. Interestingly, the major of comments (110) were directed at the T-chart representations, which use the four-box and SR.BID method of representation to evaluate analogies.

7.4.2.6 Evaluation of Study I.2

Quantitative value and rank-order questions are evaluated quantitatively, by looking for general trends in numerical results using descriptive statistics. Because of the small

sample size, in particular with respect to certain majors, inferences based on measures of statistical significance are avoided.

Qualitative data collected from open-ended questions, both from survey data and from take-home final data, are assessed using a grounded categorization scheme, with separate categorization schemes developed for each question as the data directs. Categories are names such that they retain an abstract notion of the semantics associated with the comments. Frequency of the each category is measured with respect to the all answers associated with each question; the more frequently a category occurs, the more weight and meaning is assigned to that category relative to the question.

In the case of final exam categories, two further categorization schemas are used: one assigns each category to a subject: either the four-box method, SR.BID representations, or the T-chart; and the other assigns each category as either positive criticisms or negative criticisms. These additional categorizations allow discrimination tools, and provide a high level view of the overall trend in positive versus negative comments.

7.4.3 Results of the Study

For the rank order and value-scale data, answers were tallied in a spreadsheet and for general trend analysis averages were observed for the group and by major. The following table 7-4, table 7-5, and table 7-6 provide averages for each survey question, by major. Recall again that the first question was rank-order from 1 to 5 without repetition, while the second uses a 5 to 1 value scale with repetition. While the number of responses is not sufficient to draw significant statistics, especially with respect to majors, some trends are worth noting. With respect to the first question, the skill that appears to be most important to students practicing BID is *understanding the biology*, while *finding biology*

and *defining the problem* rank next most important. This is particularly interesting, as comparatively more research effort is being placed on finding biological analogies. This table provides some supporting evidence that the challenge of defining a problem is at least as important as that of search in biologically inspired design.

Table 7-7. Survey question 2, important skills

Averages: Question 2, Skills						
N	Major	Finding bio	Understanding bio	Analogizing	Defining problem	Applying mechanism
3	BIO	3.0	1.7	4.3	2.7	3.3
7	BME	3.4	2.4	3.9	2.1	3.1
6	ISYE	1.8	3.0	3.8	3.5	2.8
7	ME/PTFE	3.3	2.1	3.7	3.1	2.7
23	Total	2.9	2.4	3.9	2.9	3
SCALE: rank 1(most) to 5(least)						

Table 7-8. Survey question 3, problem concepts.

Averages: Question 3, Problem Concepts								
N	Major	Operational Environment	Material constraints	Manufacturing constraints	Function	Performance Criteria	Cost	AVG
3	BIO	4.3	4.0	3.0	4.7	3.7	3.0	3.8
7	BME	3.6	3.0	3.0	5.0	3.7	2.4	3.5
6	ISYE	3.7	3.7	2.7	4.5	3.8	3.2	3.6
7	ME/PTFE	3.3	4.1	3.4	5.0	4.1	3.0	3.8
23	Total	3.6	3.7	3	4.8	3.9	2.9	3.64
SCALE: 1(low) to 5(high) with repetition								

Trends in the rank ordering of the importance of different concepts in problem definition reinforce the idea that functions play a key role, while the other concepts of the four-box method (operating environment, performance criteria, and materials constraints) appear to be important as well relative to, for example, cost or manufacturing constraints. It is difficult to ascertain if this rank ordering would hold in general, or if training on the four-box method directed student focus.

Table 7-9. Survey question 4, SR.BID use

Averages: Question 4, SR.BID Use								
N	Major	Define problem constructively	Communicate problem to others	Define bio solutions constructively	Communicate about biological solutions to others	Understand th accuracy of analogy	Communicate why your analogy was good/bad	AVG
3	BIO	2.0	2.7	2.0	2.0	2.0	2.0	2.1
7	BME	4.1	3.6	3.9	3.3	2.9	3.4	3.5
6	ISYE	3.3	3.3	3.0	3.0	2.5	2.7	3.0
7	ME/PTFE	4.6	3.4	4.9	3.0	3.9	3.6	3.9
23	Total	3.783	3.348	3.696	2.957	2.957	3.087	3.30
SCALE: 1(low) to 5(high) with repetition								

Question four seeks to directly address the relative differences in perceived value of SR.BID and the four box method with respect to the design activities. This finding suggests the SR.BID and four-box method were most useful for defining problems and biological solutions constructively, and that they were less useful for communicating and developing analogies. While the number of samples across majors is low, especially for biology majors, it is worth noting that biology majors ranked the utility of SR.BID on average 2.1, much lower compared to engineers who ranked the value on average at 3.9.

The evaluation of the category/sub-category analysis of the survey data is provided in Appendix J for completeness. It neither directly supports nor refutes the main hypothesis, but may be of interest to practitioners and instructors in biologically inspired design.

The final exam data provided a large volume of open-feedback comments with respect to SR.BID, the four-box method and the T-chart method of analogical evaluation. All data relevant to any of these concepts was transcribed to a spreadsheet as either an sentence, or as a complete paragraph depending on the complexity of the concept being expressed. Each idea was then tagged with a subject, either four-box method, SR.BID, or the T-chart method, and a short phrase. For example the following sentence was categorized as “SR.BID – clarifying problem”:

“Furthermore the SR.BID techniques for problem decomposition allow for a very detailed emphasis of each aspect of the problem, thus narrowing the problem and providing a more clear idea of the essential aspects of the problem that need to be addressed.”

Some paragraphs expressed multiple ideas which could not be cleanly divided, in which case multiple tags were used. For example the following paragraphs was tagged as (1) “4-Box – problem breakdown”, (2) “SR.BID – translate biological solutions to engineering design,” and (3) “SR.BID – deconstruction problems.”

“An example of a success in applying a BID method was in the problem breakdown for our final project, which helped us to successfully identify and address nearly all pertinent issues using the four box method. SR.BID also helped us translate our found solutions to the engineering domain by deconstructing them and clarifying which components we could use and why.”

Tags were sorted according to their subject. Phrases were then aggregated into similar concepts, and assigned to a higher level category. For example “define requirements,” “problem definition,” and “problem specification” were assigned to the higher level category “define/specify/clarify problem”. Each high level category was assigned as being either a positive or negative comment. The number of comments in each category

were then tabulated, indicating general trends in the comments with respect to the four-box method, SR.BID and the T-Chart method of analogical evaluation.

Table 7-7 shows the number of comments by category for all comments related to the subject of the four-box method. The left had side of the table shows the “positive comments” while the right had side shows the “negative comments.”

Table 7-10. Comments, count by category for four-box method.

Positive Comments	29	Negative Comments	10
Define/specify/clarify problem	8	Decrease/Limit creativity	3
Breakdown, problem	6	Limited to a single environment	2
Focus	2	Confusing, categorizing concepts	1
Organize data/knowledge/problem	2	Confusing, redundant	1
Search, aid	2	Confusing, specification v performance criteria	1
Understand, system	2	Difficult to learn, to use initially, different at first	1
Analogy, matching	1	Increased workload	1
Direct inquiry	1		
Easier than another system	1		
Evaluate, problem	1		
Understanding, SR.BID	1		
Useful, operational environment	1		
Visualization	1		

First, we note that positive comments outweigh negative comments by nearly a 3-to-1 ratio. Recall the assignment asks students to “describe one example of success and one example of failure to apply the techniques learned in class” from which we might expect an equal number of positive and comments would be found, from a “value-neutral process.” The relatively higher number of positive comments suggests a either a higher perceived value or, more cynically, at least a higher perceived value on commenting on successes. The first four of the comments on the positive side provide *strong evidence*

for the value of the four-box method of problem specification: in particular that, as intended, *it provides students with a greater capability to define/specify/clarify, breakdown, organize, and otherwise grapple with complex design problems.* On the downside, several students felt that the technique limited their creativity or was otherwise confusing. That structure is perceived to limit creativity is not entirely unexpected, in particular in an environment that emphasizes creativity.

With respect to SR.BID representations in general, Table 7-8 shows the breakdown of positive and negative comments. Unlike with the four-box method, comments about the SR.BID method of representation – which was used for more than just problem specification – suggests a greater amount of confusion for usability. This may be a result of the way in which SR.BID was presented, as a higher level abstraction of knowledge representation, divorced from a specific task for which it could be used. Note that 8 of the 22 negative comments (the top row) are concerned with the perceived difficulty of learning or using initially SR.BID initially. Study 5.1 shows that, at least when contextualized in the task of problem definition using the four-box method, students were very successful in a short period of time. This category also implies that the problem exists initially, but is overcome with experience.

Table 7-11. Comments, count by category for the SR.BID representation.

Positive Comments	28	Negative Comments	22
Define/specify/clarify problem	4	Difficult to learn, to use initially, different at first	8
Breakdown, problem	3	Confusing, specification v performance criteria	2
Organize data/knowledge/problem	3	Incomplete/unable to specify problem	2
Analogy, identification/selection	2	Confusing, non-physical problems	1
Breakdown, system	2	Confusing, non-physical/non-mechanical systems	1
Helps framing	2	Confusing, performance criteria may not apply	1
Analogy, development	1	Decrease/Limit creativity	1
Analogy, matching	1	Difficult to narrow down problem	1
Analogy, translation	1	Difficult to use	1
Communciation, interdisciplinary	1	Difficult to use with solutions	1
Communication, aid	1	Increased workload	1
Compare, systems	1	Limited scope	1
Easier in later stages/for more well defined problem	1	Not used	1
Evaluate, problem	1		
Focus	1		
Process, requires iteration	1		
Understand, system	1		
Understanding, problem	1		

Finally, I include here comments about the T-chart method for analogical evaluation. Although a complete analysis of the T-chart method is out of scope for this dissertation, I think the amount and type of comments about the method collected in the reflections speak strongly to the *value of the SR.BID representation schema when it is contextualized in a tool and a design task*. Table 7-9 provides a summary of the comments on the T-chart method of evaluations. When situated in a tool and task, the number of positive comments outweighs the negative comments by a 2-to-1 ratio. What is particularly encouraging is that *30 of positive comments collected through open feedback, are directed at the core value proposition of the tool: the use of SR.BID for*

analogical identification, selection and evaluation.

Table 7-12. Comments, count by category for the T-chart method.

Positive Comments	74	Negative Comments	36
Analogy, evaluation/measure strength	20	Ad hoc/specious/not-genuine/immaterial	4
Analogy, identification/selection	10	Redundant	4
Visualization	5	Confusion, relevance of differences/similarities	3
Analogy, identifies similarities/differences	4	Difficult to use	3
Easy to use	4	Not useful	3
Analogy, translation	3	Confusing, measuring similarity	2
Analogy, understanding	3	Confusion, evaluation of partial matches	2
Focus	3	Did not use/did not use on all	2
Analogy, validation	2	Confusion, hard to identify correspondence	1
Communication, consensus	2	Confusion, hard to select proper phrase	1
Communication, to external groups	2	Confusion, one-to-one evaluation not effective	1
Ensure a BID solution	2	Creates criticism	1
Organize data/knowledge/problem	2	Depends on problem understanding	1
Analogy, selection	1	Felt forced	1
Breakdown, problem	1	Inferior to intuition	1
Breakdown, system	1	Leads to unnecessary criticism	1
Communication, aid	1	Needs quantified measures	1
Communication, common vocabulary	1	Needs to be complete	1
Creativity, enables	1	Over simplified	1
Eliminates bias	1	Requires multiple solutions	1
Ensured completeness	1	Superficial	1
Flexible	1		
Knowledge integration	1		
Search, aid	1		
Understanding, system	1		

On the other hand, the negative side of the chart suggests that the tool is not without its flaws. The T-chart tool at the cost of being simple to use, provides only a superficial means for comparing analogies using SR.BID, which may lead to a number of the comments. An attempt to compare complete SR.BID models, which would enable

structural (relational) comparison as traditional theories of analogy suggest are necessary for rich analogical evaluation, would lead to complex modeling framework and tool, that while more thorough, would also have a much greater learning curve and investment cost. While it is not yet clear what balance needs to be struck there, it does appear that SR.BID is again providing some incremental value at a relatively low cost. It is worth noting that instructors are continuing to use the T-chart method of analogical evaluation in 2013, in my absence.

7.4.4 Summary of the Study

This study attempts to ascertain the perceived value of the four-box method of problem formulation, as well as the perceived value of the underlying SR.BID representation schema. Survey data suggests that in the context of BID students view the definition of the design problem as equal in importance to searching for biological system, and second only to the importance of understanding the underlying biological systems. In addition the survey data indicates the SR.BID and four-box method of representation are provide value to the students for the tasks of both problem and solution definition.

This data is further supported by findings from student reflections collected in take home final exams, in which instructors asked open-questions about student successes and failures with various tools and techniques learned in class. Although students were free to comment on anything in class, most students reported on their experiences with the SR.BID representation and four-box method. From these undirected student comments, I assert that students found the four-box method of problem specification value in general (citing a three-to-one ratio in positive versus negative comments), and specifically

valuable for the task for which it was designed, problem definition. Secondly, and although not the thrust of this study, student comments show that SR.BID, *when contextualized in a tool and a design task*, is a valuable tool both for problem definition and for analogical evaluation.

7.5 Claim 4

The four-box method derived from SR.BID can be successfully used by all students in the BID classroom for accurately representing problems in the context of BID.

7.6 Study I.3: A Web-based Implementation of SR.BID

Subsequent to the 2006 and 2007 studies, I implemented a prototype system, DANE, in class to scaffold student design tasks. Because of the diversity of design problems and because of the number and diversity of biological analogues that might be useful for solving any one of those problems, providing a tool with a sufficient number of biological analogues available is critical to the perceived value of these systems. While providing a large number of analogues is critical, it must be balanced against the effort of knowledge engineering required to provide these analogues.

DANE representations used complete SBF representations, which provide inferential power over complex design tasks, both for humans (Helms et al) and computer agents (Goel, Bhatta and Goel), with a corresponding cost in terms of knowledge engineering. Student feedback from the DANE implementation established that more biological systems would dramatically increase the value of these systems. AskNature, as an example of another system available to the systems, with tens of thousands of biological sources was used frequently. AskNature, uses lightly structured data similar in form to

Wikipedia, curated by a team of professionals, with an overlying functional (strategies) ontology used for navigation. Each technological system must carefully consider the cost of knowledge engineering versus the utility provided by the underlying representation; typically the more capability a representation provides, the more difficult the task of knowledge engineering, and the more costly the cost-per-unit of bringing in new content. This is further complicated in BID because the number of possible analogical sources is huge (billions of potential species, each capable of performing many functions). This leads to the formulation of my fourth research problem.

7.6.1 Research problem 4

From observations of systems implemented to support BID, we know there is a tradeoff between representational complexity, cost of representation and potential value/tasks supported. Existing support systems do not yet provide a sufficient return on investment such that system builders are able to get widespread adoption. How might these systems acquire structured data on thousands or tens of thousands of biological source analogues that would be necessary to drive value to the design users?

7.6.2 Study Motivation

I demonstrated earlier that SR.BID and the four-box method provide useful problem and solution description support in the context of biologically inspired design. This study extends the value of these tools, by making them available through a distributed web-based platform. The study demonstrates how a designer can translate their design projects into sharable and well-structured knowledge representations on the web with little additional effort.

7.6.3 The Functionality of the SR.BID Web Application

This section describes the SR.BID Web application in detail. The application is designed as a collaborative design environment, where design teams may share unstructured information on their design project, as well as create structured four-box representations for design problems and biological solutions. As the number of biological solutions grows, search capability enables designers to match biological solution with design projects using the four-box specification. Readers not interested in the technical specifications and functionality of the Web Application, may wish to skip this section.

The SR.BID Web application runs a web application stack consisting of an Apache 2.2.11 web server, using PHP 5.3.0 for dynamic content generation and a MySQL 5.1.36 database. Formatting is generally static, with little accommodation for different screen resolutions, using CSS. Although most machines now operate in resolutions significantly greater than 800 pixels, displays or windows less than 800 pixels wide may experience usability issues. The client side makes use of, javascript and jquery 1.8.23 to manage client side interactions, and AJAX (managed with jquery) for asynchronous data retrieval. All major browsers as of the time of release were supported except Internet Explorer. A database ERR diagram is included in Appendix K for reference.

At the highest level, designers may (1) manage their account, (2) create new and edit to design projects, (3) create and edit biological system information, (4) learn about the SR.BID Web Application and SR.BID representations, and (5) learn about the designers and intent of the site. Figure 7-7 shows the menu for these high-level options. Account, Learn and About options are largely administrative and uninteresting from a research perspective. I will review the design project and biological systems options in more

detail.



Figure 7-7. Web application high-level menu.

7.6.3.1 *Design Project*

A user may add a new design project or edit an existing project. A list of links to existing projects with which the user is associated, see figure 7-8, is displayed on the users left when they log in, so they may navigate directly to editing an existing project.

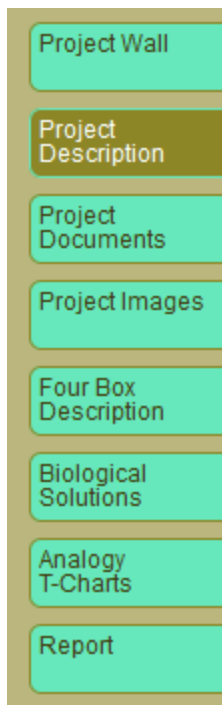


Figure 7-9. Project options.



Figure 7-8. Member list of existing projects

The main options displayed for each project, (figure 7-9) are to: post comments on the project wall, add/edit/delete the project description, project documents, project images, a four-box description, associate biological solutions, analogy T-charts, or generate a report. Of these functions, the project wall, analogy T-charts and report generation were simply placeholders for future functionality and not available to designers at the time of implementation. The project description function provided for free-text entry of a project description (not shown). The project documents function enabled the designer to add any file to the project, and provide a short description of the file for

team members. Figure 7-10 shows the project document entry screen. The top frame provide the capability for entering new documents, while the lower frame lists existing documents associated with the project, in this case, two sample documents about plaque removal. The “view” button allows team members to directly view the documents, if the documents are supported by a in a browser-based viewer.

Project Documents	
Plaque removal with a novel manual toothbrush (X-Active) and the Braun Oral-B 3D Plaque Remover	View Save Delete
An advanced toothbrush with advanced plaque removal efficacy	View Save Delete

Figure 7-10. Project document entry screen.

Project Images are supported using the same basic structure, as shown in figure 7-11. Images are shown as thumbnails, which can be pulled up in a separate window at full size, by clicking on the View button.

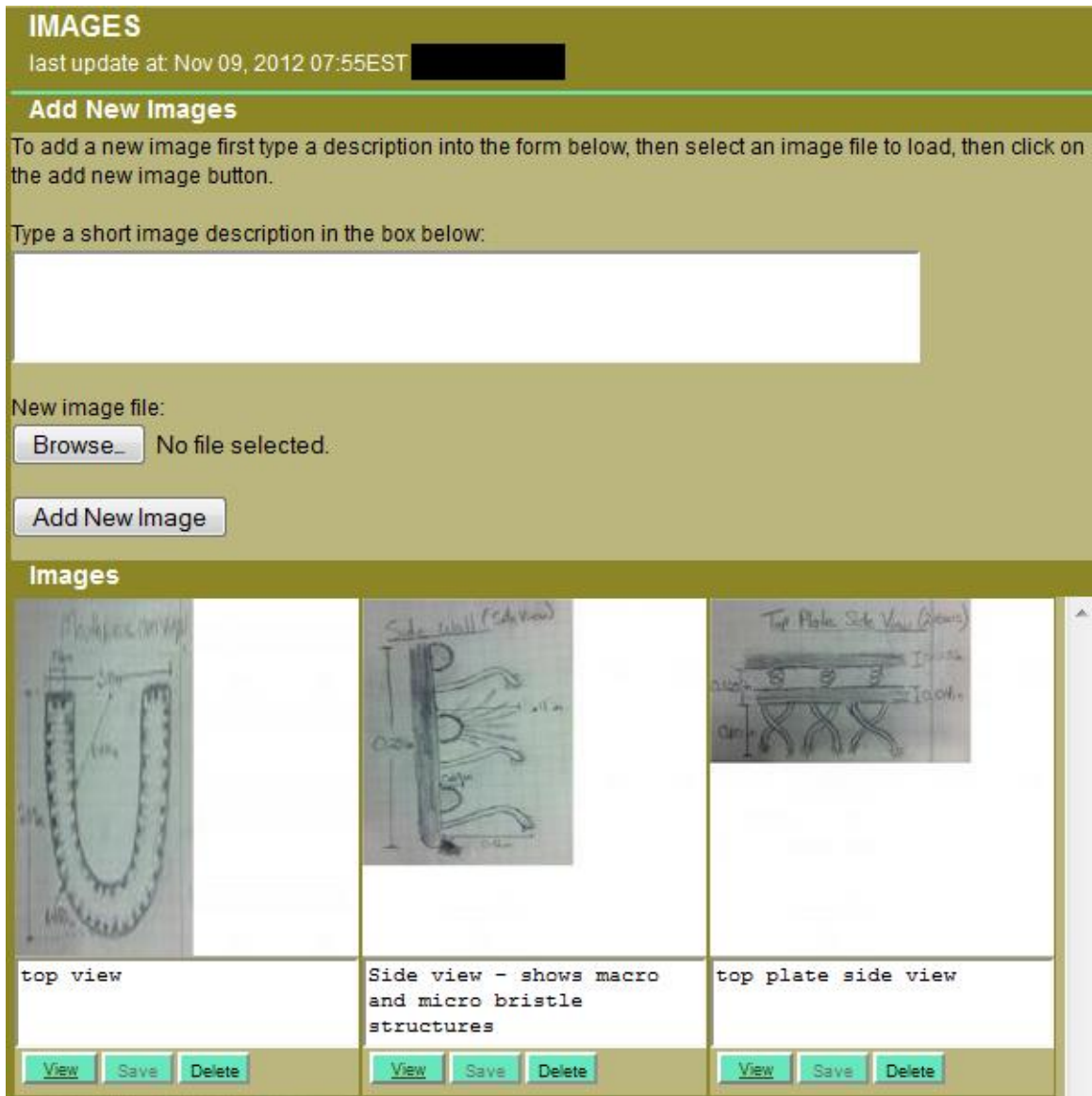


Figure 7-11. Project image entry screen.

The four-box function allows a four-box model of the problem to be specified. Figure 7-12 provides a partial screen-shot of the four-box editor in use. This screen shot shows the operational environment and function frames of the four-box model. For each frame, a design may enter a concept using for instance the “add operating environment” or “add function” buttons, shown in this screen shot.

FOUR BOX DESCRIPTION

last update at Nov 11, 2012 10:32EST Michael Helms

Operational Environment	Operational Env. Type	Options
Mouth, teeth, tongue, gums	Location	<input type="button" value="Delete"/>
Saliva, toothpaste, bacteria, dental plaque	Condition	<input type="button" value="Delete"/>
Adults, Children, Dentists & other medical profe	User	<input type="button" value="Delete"/>
<input type="button" value="Add Operating Environment"/>		

Function	Engineering Function Type	Options
Safe effective removal of dental biofilm	Remove/Block	<input type="button" value="Delete"/>
Dislodge plaque and other food particles	Remove/Block	<input type="button" value="Delete"/>
Prevent periodontal inflammation	Protect	<input type="button" value="Delete"/>
Prevent oral irritation and ulceration	Protect	<input type="button" value="Delete"/>
Deliver agents to the tooth surface	Provision/Supply	<input type="button" value="Delete"/>
Stimulate gum growth	Support	<input type="button" value="Delete"/>
<input type="button" value="Add Function"/>		

Figure 7-12. Partial screen shot of four-box entry screen.

When the “add” button is clicked the user is provided with a line to enter the new concept, for example a new operational environment. As text for the new concept is entered, the database is searched, and any matching concepts are displayed. In this way, if the concept (for example “Australian Desert”) is already entered, the user may select it without having to type in the entire concept. This also enhances the potential for future matches, when searching for biological sources. Figure 7-13 shows a new operational environment being added.

Operational Environment	Operational Env. Type	Options
Mouth, teeth, tongue, gums	Location	Delete
Saliva, toothpaste, bacteria, dental plaque	Condition	Delete
Adults, Children, Dentists & other medical profe	User	Delete
Des	--Select Oper. Env.--	
cannot be a desert		
Desert conditions		
Desert		
Australian desert		
Sandy Loess Plains & Stony Desert Areas		
Engineering Function Type		Options
	Remove/Block	Delete
	Remove/Block	Delete
	Protect	Delete

Figure 7-13. Screen capture from adding a new operational environment.

After the user enters a free-text description, they must also specify a sub-types description, as specified in the complete SR.BID ontology. Thus for operational environment, the user is prompted with a drop-down box to select location, condition, condition value, etc. Forcing the user to enter a sub-type provides a prompt for users to consider all of the available sub-types, hopefully providing a deeper of consideration of the entire problem model. Figure 7-14 shows a screen capture from a user selecting from the drop-down list of operational environment sub-types.

FOUR BOX DESCRIPTION


last update at: Nov 11, 2012 10:32EST Michael Helms

Operational Environment	Operational Env. Type	Options
Mouth, teeth, tongue, gums	Location	<input type="button" value="Delete"/>
Saliva, toothpaste, bacteria, dental plaque	--Select Oper. Env.--	<input type="button" value="Delete"/>
Adults, Children, Dentists & other medical profe	Location	<input type="button" value="Delete"/>
<input type="button" value="Add Operating Environment"/>		
<ul style="list-style-type: none"> Condition Condition Value User Reactive Entity Time Period System 		
Function	Engineering Function Type	Options
Safe effective removal of dental biofilm	Remove/Block	<input type="button" value="Delete"/>
Dislodge plaque and other food particles	Remove/Block	<input type="button" value="Delete"/>
Prevent periodontal inflammation	Protect	<input type="button" value="Delete"/>
Prevent oral irritation and ulceration	Protect	<input type="button" value="Delete"/>
Deliver agents to the tooth surface	Provision/Supply	<input type="button" value="Delete"/>
Stimulate gum growth	Support	<input type="button" value="Delete"/>
<input type="button" value="Add Function"/>		

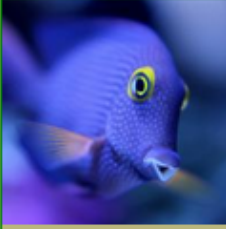
Figure 7-14. Screen shot of sub-type selection for operational environment.

Designers are also able to view biological sources associated with a project. In figure 7-15, two associated biological sources are shown, the Camel nose and the Ctenochaetus. All information associated with the biological source is accessible from this frame, including descriptions, images, reference documents, and the four box model of the biological source (shown in the figure 7-16). Additionally, as shown in figure 7-17 designers are able to document why they felt this analogy was relevant to the project, and why they accepted or rejected the analogy for the final design. This functionality was added as a temporary measure, in lieu of the T-chart which was not supported in this release.

ASSOCIATED BIOLOGICAL SOLUTIONS
last update at: EST



Camel Nose



Ctenochaetus



Camel Nose

Descriptions **Four Box Model** Analogy Evaluation

- Solution Description

Respiratory water loss is unexpectedly low in camels that were subjected to severe water deprivation, and in fact so low that the results were incompatible with the view that exhaled air is saturated with water vapour. The respiratory air of camels and several other large mammals may be exhaled at or near ambient rather than body core temperature (Langman et al. 1978, 1979). This cooling of the exhaled air results in substantial savings in water; camels can further reduce the respiratory water loss by extracting water vapour from the exhaled air, resulting in the exhalation of air at less than 100 % relative humidity (r.h.).

- Solution Images

Camel nose Camel turbinates with very high surface area

- Solution References


K. Schmidt-Nielsen, R. C. Schroter, A. Shkolnik, Desaturation of Exhaled Air in Proceedings of the Royal Society of London. Series B, Biological Sciences, Vol. 1184 (Mar. 11, 1981), pp. 305-319 Stable URL: <http://www.jstor.org/stable/35545>

[View File](#)

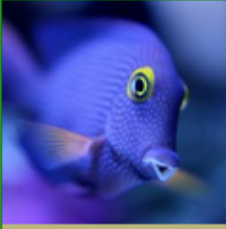
Souilem Ouajd, Barhoumi Kamel. Physiological Particularities of Dromedary (*Dromedarius*) and Experimental Implications. Scand. J. Lab. Anim. Sci. 2009 V

Figure 7-15 Biological analogue source descriptions, Camel Nose.

ASSOCIATED BIOLOGICAL SOLUTIONS
last update at: EST



Camel Nose



Ctenochaetus

Camel Nose

[Descriptions](#)
[Four Box Model](#)
[Analogy Evaluation](#)

- Solution Four Box Model

Operating Environment

Description	Type
Desert	Location
Relative humidity between 25-50%	Condition Value
Temperature between 26C-40C	Condition Value

Functions

Description	Type
Retain water	Provision/Supply
Desaturate air	Change Condition
Cool brain	Change Condition
Regulate bloodflow to nasal passages	Maintain Magnitude

Specifications/Constraints


Description	Type
1000 square centimeters of exposed surface area	Property/Value
Scroll-like turbinates	Shape

Performance Criteria

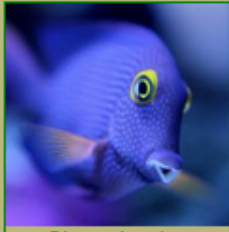
Description
3.42 kJ of heat removed per minute
75% recovery of exhaled moisture

Figure 7-16. Biological analogue four-box description, Camel Nose.

ASSOCIATED BIOLOGICAL SOLUTIONS
last update at: EST



Camel Nose



Ctenochaetus

Ctenochaetus

Descriptions Four Box Model **Analogy Evaluation**

Save Analogy Evaluation

- Description of relevance to project

The original purpose of the knobby stripe is to allow the fish easier access to feed; however, the purpose used for the B.I.T.E. is to better clean the teeth.

- Justification of inclusion/exclusion

Removal of biofilms from surfaces are similar functions.

Figure 7-17. Analogy rationale, Camel Nose.

7.6.3.2 *Biological Systems*

The first function associated with biological systems, is the ability to add a new biological system to the database. Using a very similar layout as with projects, users are prompted to enter a free-text biological system description, a four-box description of the biological system (a partial four-box description for a scorpion is shown in figure 7-18), references and images.

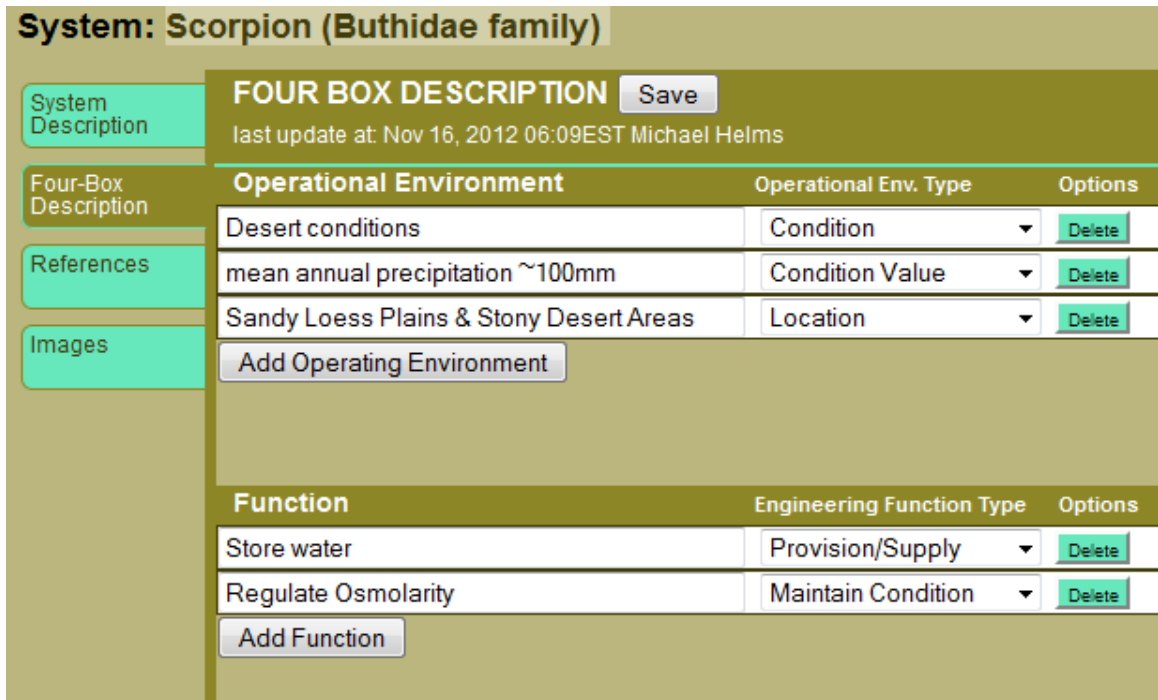


Figure 7-18. Screen shot of the four-box model for a biological system, Scorpion.

The second function associated with biological systems is the ability to browse/search the database of biological systems. In browsing mode, a complete query may be constructed and executed to retrieve a list of ordered matches. Figure 7-19 shows the query construction framework.

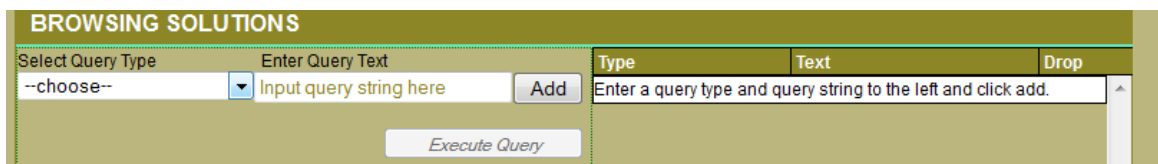


Figure 7-19. Biological system query construction.

A user may select a query type based on concepts in the four-box model, based on querying the text descriptions, or based on a search of all associated text, as shown in

figure 7-20. Once a category is selected the user enters a term. For example the user may select “Function” from query teyp, and then enter the term “water”, which will bring up a pick list of all functions that contain the query term water, as in Figure 7-21. The user may select from the pick list, for example by selecting “retain water”, which will, when the query is executed select all systems with the function “retain water.” Or the user may leave the more general term “water” which will then, when the query executes, return all systems with functions that contain the word “water” (e.g. harvest water, protect water from scavengers, etc.) Note the “complete list” button is a temporary feature for convenience, and will execute a query that will return ALL biological sources in the database.

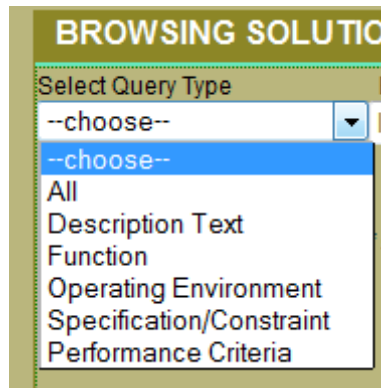


Figure 7-20. Screen shot of query type drop down menu.

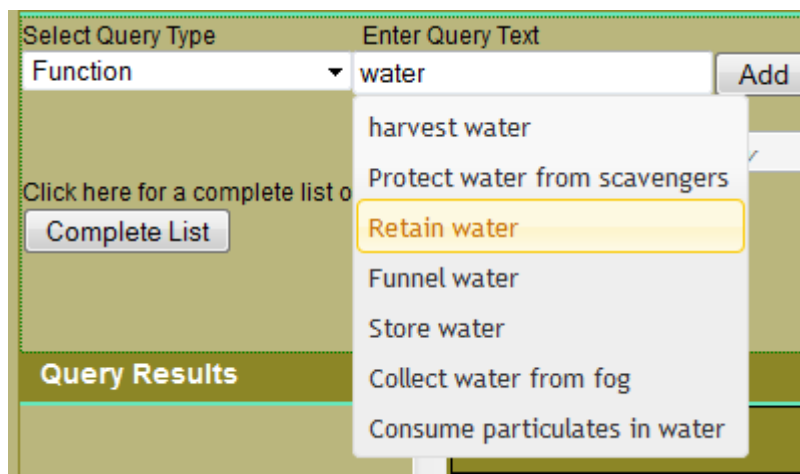


Figure 7-21. Drop down pick list for functions containing the query term “water.”

Multiple query terms may be added together, to form a compound “OR” query, for example to return all biological sources that have a function which contains the word “store water” OR “retain water” or “harvest water” OR where the operational environment is “desert”. Figure 7-22 shows these query terms. The drop button allows the user to selectively drop terms from the query.

Type	Text	Drop
function	Store water	Drop
function	Retain water	Drop
function	harvest water	Drop
OE	Desert	Drop

Figure 7-22. Compound “OR” query, using multiple terms and types.

When the query is executed, a list of biological sources that match the query is returned, as a list of annotated thumbnail images. This allows users to browse the return query by both name and image. Figure 7-23 represents the image return set for the water and desert conditions query, above.



Figure 7-23. Biological system queries return annotated images.

When one of the query images or names is clicked on, the relevant information associated with that biological source is retrieved. Figure 7-24 shows the description, image and reference information associated with a biological source, in this case for the Thorny Devil. Figure 7-25 shows the four-box model associated with the same.



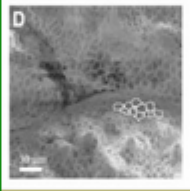
Thorny Devil (Moloch Horridus)

[Descriptions](#)
[Four Box Model](#)
[Add to Project](#)

System Description

The Thorny Devil is an ant eating lizard from the Australian desert. It is known for its spined back that acts as a deterrent to potential predators and facilitate moisture collection in the arid climate. The Thorny Devil uses cutaneous capillary action to collect, funnel, and imbibe water in arid conditions. The cutaneous capillary system is a skin-level circulatory system made of a system of grooves. The groves are sub-scalar channels tucked into its interscalar channels with an overlapping shelf at the edge of the scales. Channels radiate from the Moloch's spines to facilitate collection of dew and droplets from rainfall. The spines act as collection foci for moisture intake. When water falls on the Moloch it runs down the microgrooves on its back. The capillary force supports a pressure head of 10cm water to facilitate flow of the water to the animal's mouth. The

System Images

		
Thorny devil image	Thorny devil scales	Thorny Devil small scales

System References

Wade C. Sherbrooke, Andrew J. Scardino, Rocky de Nys, Lin Schwarzkopf. Functional morphology of scale hinges used to transport water: convergent drinking adaptations in desert lizards. *Zoomorphology* (2007) 126:89–102

[View File](#)

Thorny Devil Lizard: Information on Australia's thorny devil lizard (Moloch horridus). Text and photos by Wade C. Sherbrooke. Reptiles.com

Figure 7-24. Description associated with a biological system, Thorny Devil.

Thorny Devil (Moloch Horridus)	
Descriptions	Four Box Model
Add to Project	
System Four Box Model	
Operating Environment	
Description	Type
Australian desert	Location
Arid climate	Condition
Functions	
Description	Type
Deter predators	Protect
Collect moisture/dew	Import
Funnel water	Guide
Collect rain	Import
Facilitate capillary action	Change Condition

Figure 7-25 Four-box model of biological system, Thorny Devil

When a biological source is clicked on, the user also has the ability to add the biological source to one or more projects with which that the user is associated. Figure 7-26 shows the ability to select from among any project that the user has access to, and also shows in the bottom frame an project that the user has access to that is already associated with the selected biological source. This is currently the only way of searching for a biological source, and associating it with a project. Future releases should consider offering search & association points more closely linked to the project, for instance a function to automatically find and return biological sources that match across the four-box description of the design problem.

The screenshot shows a web application interface for the 'Thorny Devil (Moloch Horridus)'. At the top, there are three tabs: 'Descriptions', 'Four Box Model', and 'Add to Project'. Below the tabs is a section titled 'Add this solution to a design project'. This section contains the following text: 'Select a design project below, and click ADD to add the solution to your design project. You will then be able to access this information when working on your design project.' Below this text is a list of radio buttons with the following options: 'hierarchical structures', 'Team 8: Signal Seed', 'Bio-Inspired Toothbrush Extraordinaire (B.I.T.E)', 'Group 2 - Pipecone', and 'Gongbo Example Project'. Below the list is a button labeled 'Add Solution to Project'. At the bottom of the interface is a section titled 'Projects to which this solution has already been added', which is currently empty.

Figure 7-26 Adding an existing biological system to a design project.

7.6.4 Study Methodology

The SR.BID Web Application is technological implementation of the SR.BID representation intended as a proof-in-concept of the applicability of SR.BID as a means for large-scale data capture for use in biologically inspired design. The application is made available to students in the BID class, to be used at their discretion. In this study, I examine the apparent incremental cost to students of using the SR.BID Web Application through empirical observations of transactions entered into the system. These observations can be used for assessing the general utility and cost of such a system for larger scale implementations.

7.6.4.1 *Research question I.3*

What is the additional incremental cost to a design team for using SR.BID and the four-box method in a sharable, distributed (e.g. web-based) platform?

7.6.4.2 *Hypothesis I.3*

The SR.BID for Web Application can be used to support designers for problem definition, biological analogy building, and analogical search and evaluation, over a distributed team-based platform in the context of BID, with minimal additional investment over current assignment workload.

7.6.4.3 *Context and Participants I.3*

The SR.BID Web Application is deployed in week 12 of the 2012 BID@GATech class (see Study 5.2 for details on that class). Students are encouraged to use the tool on their own time at their own discretion. Use is entirely optional, and has no bearing on student grades or assignments.

7.6.4.4 *Execution of Study I.3*

The SR.BID Web Application is a web-based application developed for use on all standard browser technology. The application functionality is reviewed with instructors and the class during a 20 minute training session in week 12 of class. The application is framed for students as a future research tool, and students are asked by researchers to voluntarily use the system to enter problem definition information, and biological source information.

Students that decide to use the tool create a user ID and password. Thereafter all student data entry transactions are logged against that ID in the application database. Data extracted from the transaction logs in the database are evaluated to determine the

number, type and timing of transactions. The content entered by student designers is also collected.

7.6.4.5 Data I.3

Two kinds of data are collected, transaction log information and content. In total 7 out of 34 students registered in the system. Of the 7 users, four users did not execute any transactions other than to register. One registered user entered only a single transaction, to add a new project. No additional transactions were made related to that new project.

The two remaining users (#6 and #7) contributed meaningful amounts of data to the system. Table 7-10 and table 7-11 summarize the transactions, scope and timing (to the nearest minute) of their work.

Table 7-13. SR.BID Web Application transaction data, User #6.

USER ID	DATE & TIME	TRANSACTION TYPE	TRANSACTION
6	11/8/2012 19:52	NEW PROJECT	Added Project: The Signal Seed
6	11/8/2012 19:56	NEW REFERENCE	Added Referenece: Project document, final report.pdf
6	11/8/2012 19:57	NEW REFERENCE	Added Reference: Project document, materials assessment.pdf
6	11/11/2012 0:05	NEW REFERENCE	Added Referenec: Project image, images of prototypes 1 & 2
6	11/11/2012 0:07	NEW BIOLOGICAL SYSTEM	Maple seed (samara)
6	11/11/2012 0:07	NEW PROJECT	Maple seed (samara)
6	11/11/2012 0:13	NEW REFERENCE	Added reference: samara seed distributio.pdf
6	11/19/2012 11:29	NEW REFERENCE	Blank entry
6	11/19/2012 11:30	NEW REFERENCE	Added Reference: Project document, QA.pdf
6	11/19/2012 11:30	DELETE EXTERNAL FILE	Deleted blank entry

Table 7-14 SR.BID Web Application transaction data, User #7.

USER ID	DATE TIME	&	TRANSACTION TYPE	TRANSACTION
7	11/9/2012 10:38		NEW PROJECT	Added project: BITE
7	11/9/2012 10:50		NEW REFERENCE	Added refernce: Plaque removal with a novel manual toothbrush (X-active) and the Braun Oral-B 3D Plaque Remover.pdf
7	11/9/2012 10:51		NEW REFERENCE	Added refernce: An advanced toothbrush with improved plaque removal efficacy.pdf
7	11/9/2012 10:52		NEW REFERENCE	Added refernce: A Clinical Evaluation of the Plaque Removal Efficacy.pdf
7	11/9/2012 10:54		NEW REFERENCE	Added refernce: Improved Plaque Removal Efficacy.pdf
7	11/9/2012 10:55		NEW REFERENCE	Added refernce: Evaluation of toothbrush wear on toothbrushing efficiency.pdf
7	11/9/2012 10:57		NEW REFERENCE	Added refernce: ToothbrushTechnology (1).pdf
7	11/9/2012 10:58		NEW REFERENCE	Added refernce: Dental Biofilm thesis.pdf
7	11/9/2012 10:59		NEW REFERENCE	Added refernce: Biofilm fluid mechanics.pdf
7	11/9/2012 11:02		NEW REFERENCE	Added refernce: Antifouling adaptations of caridean shrimps.pdf
7	11/9/2012 11:03		NEW REFERENCE	Added refernce: Paratya anatomy.pdf
7	11/9/2012 11:16		NEW OE	Added OE: Mouth, teeth, tongue, gums
7	11/9/2012 11:16		NEW OE	Added OE: Saliva, toothpaste, bacteria, dental plaque
7	11/9/2012 11:16		NEW OE	Added OE: Adults, Children, Dentists & other medical professionals
7	11/9/2012 11:16		NEW OE	Added OE: I didn't mean to put this one in but don't know how to remove it.
7	11/9/2012 11:16		NEW SPEC	Added Specification: Increased lifespan compared to other tooth brushes
7	11/9/2012 11:16		NEW SPEC	Added Specification: Durable hydrophobic materials
7	11/9/2012 11:16		NEW SPEC	Added Specification: Prevents accumulation of biofilm on bristle head
7	11/9/2012 11:16		NEW SPEC	Added Specification: Hands free, self powered device
7	11/9/2012 11:16		NEW SPEC	Added Specification: More efficient alternative to current solution
7	11/9/2012 11:16		NEW SPEC	Added Specification: Autonomous cleaning coverage
7	11/9/2012 11:16		NEW FUNCTION	Added Function: Safe effective removal of dental biofilm
7	11/9/2012 11:16		NEW FUNCTION	Added Function: Dislodge plaque and other food particles
7	11/9/2012 11:16		NEW FUNCTION	Added Function: Prevent periodontal inflammation

Table 7-14 continued

7	11/9/2012 11:16	NEW FUNCTION	Added Function: Prevent oral irritation and ulceration
7	11/9/2012 11:16	NEW FUNCTION	Added Function: Deliver agents to the tooth surface
7	11/9/2012 11:16	NEW FUNCTION	Added Function: Stimulate gum growth
7	11/9/2012 11:16	NEW PERF	Added Performance: Reliable cleaning coverage
7	11/9/2012 11:16	NEW PERF	Added Performance: Complete cleaning of all teeth in mouth
7	11/9/2012 11:16	NEW PERF	Added Performance: Eliminates the need to use a gum stimulator
7	11/9/2012 11:16	NEW PERF	Added Performance: Requires less effort than current solution
7	11/9/2012 11:16	NEW PERF	Added Performance: Mobile, hands free device
7	11/9/2012 11:19	NEW REFERENCE	Added image: Mouth Piece Top Overview.jpg
7	11/9/2012 11:19	DELETE EXTERNAL FILE	Deleted image: Mouth Piece Top Overview.jpg
7	11/9/2012 11:53	NEW REFERENCE	Added image: mouthpiece top overview.jpg
7	11/9/2012 11:54	NEW REFERENCE	Added image: side view.jpg
7	11/9/2012 11:54	NEW REFERENCE	Added image: topplate side view.jpg
7	11/9/2012 11:55	NEW REFERENCE	Added image: chewing_compressed and decompressed.jpg

7.6.4.6 Evaluation of Study I.3

Study I.3 is evaluated by assessing the number of transactions, timing of transaction and content added to the system during the time it was made available to students. The goal is to understand the feasibility in practice for student designers to use the system to create persistent, meaningful content with respect to SR.BID and four-box representations. From the two students that voluntarily used the system, I can estimate the time-cost to students to generate additional SR.BID and four-box models. This provides both instructors and tool-builders with an estimate against which to weigh the time-cost

investment against potential value to current and future students.

7.6.5 Results of Study

The first user (#6), used the system on three separate occasions 11/8, 11/11 and 11/19. Initially, the user created a new project with a short description, and added two references, a design report and a materials assessment. It took the user approximately 5 minutes to enter this data. The user completed no further transactions in that session. In the second session, the user first added an image of a prototype of their design to the project images tab. They then switched to the biological system functions, and added a biological system to the database, a samara (maple) seed with an 800+ word description detailing the biological context of the behavior of the distribution function of the seed. The user also added a reference document to the database on the behavior of the samara seed. This session lasted approximate 8 minutes. In the final session, the user returned to their project, added a reference in error (a blank reference), added a reference to their quantified project analysis, and deleted the reference which was added in error. This session lasted approximately 2 minutes. In total, using the system for approximately 15 minutes over three sessions on different days, this user created a new project, added three documents to share on the project, added one biological system including a detailed mechanistic account of a function, and added one reference paper for that system.

The second user (#7) focused more on the four-box problem description for their project. They used the system over two separate sessions on the same day, 11/9. They first added a new project with a 400+ word description and then added 10 documents to the project, all related to the nature of the design problem (cleaning teeth).. The user spent approximate 25 minutes adding this information. The user next added a complete

four-box model of the problem, including 4 operating environment concepts, 6 specifications, 6 functions and 5 performance criteria. This complete model was added in approximately 13 minutes. The user then spent 3 minutes to add and then delete an image, after which they logged out of the session. They logged back in a little more than 30 minutes later, and spent 3 minutes adding four new images, hand-drawn design sketches, to the project. In total, this user spent approximately 42 minutes building a complete problem representation, including a 400+ word description, a complete four-box model, 10 reference documents and 4 images of design sketches.

These two use-cases show that

- The initial creation of design project descriptions including free-text, document and image support can be accomplished in between 10 and 30 minutes, depending on the level of detail and amount of supporting documentation provided.
- One user shows that a biological system can be added to the database with a similar level of investment
- The creation of a rich, complete four-box model for problem description can be accomplished in approximately 15 minutes.
- In less than one hour, a student designer can document a complete design problem description using the four-box method, augmented with free-text, image, and documentation support.

7.6.6 Summary of Study

We know from studies I.1 and I.2 that student designers are capable of learning the four-box method of design problem description quickly, applying it to new design

problems with greater than 80% accuracy in most cases, and that they find value in the methodology for ordering and organizing their thinking about their design problems. Moreover there is some evidence in support of its use in analogical evaluation.

We also know from this study that individual student designers can use a web-based platform enter this information into a distributed database of problems and biological systems, and that in this prototype system they can generate meaningful, multi-model descriptions of design problems with an investment of less than 60 minutes. From this data,

7.7 Claim 5

I claim that viewed as an underlying scaffold for both tools and technology, SR.BID and the SR.BID Web Application could potentially be used for low cost, massive distributed collection of design problem and biological system information..

7.8 Future Research for the SR.BID Web Application

I did not touch upon in this research the additional value the SR.BID Web Application would provide to *search and retrieval*. Firstly using the four-box method of description for both problem and biological systems as I have done in this system, by simply specifying a problem, direct, automatic searching for biological systems is possible using the four-box concepts. Moreover, automated evaluation of these searches across all of the four-box dimensions is enable, such that more than simple functional matches are returned – but matches evaluated against environment, performance and specification concepts as well,

Secondly, this system enables problem-to-problem search as well. I believe there is

significant untapped research potential for understanding how designers might use design problem analogies to make inferences about their own design problems. Additionally to the extent that analogical design problems are associated with biological systems, one can make problem-to-problem-to-solution analogies. Such one-step-removed analogies may provide an extra leap of creativity. Whether this method of search would prove more or less effective than direct search is unknown, but the SR.BID platform provides a means for its exploration.

I will conclude this section with some informal analysis of conversations with student users that took place during the last day of class in 2012. The last day was used as an open forum for students to express their opinions about the strengths and weakness of the class, and what they think would help future classes. The feedback received during this session was unanimous in students' interest in a database of past design problems and associated biological solutions. While a database of biological solutions is useful, a database that shows *how such solutions have been applied to designs in past projects*, provides much greater traction on *understanding how to* use a biological solution to accomplish a design task. It is worth noting, perhaps, that student designers are not concerned with a database of mixed quality entries; they feel they are quite capable of separating good from bad examples, and therefore strong quality controls are not deemed necessary (at least until the number of samples was much larger).

8 CONCLUSIONS AND SUMMARY

In this dissertation I examine the phenomenon of analogical problem evolution in the context of an undergraduate design class on biologically inspired design, over the course of seven years of study. The dissertation is conducted in four phases: exploratory studies, content account development, process account development, and tool deployment. In the first phase, I identify and document the phenomenon of study: analogical problem evolution. In the second phase, I develop a knowledge model for representing and describing the phenomenon at an information processing level. In the third phase I conjecture a process account of the phenomenon, and provide evidence in support of aspects of that account. Finally, I implement a tool to support the activity of problem formulation and analogical problem evolution, and I show that not only can student designers successfully use the tool, but also that the tool can be used to scaffold the knowledge engineering effort required for robust biologically inspired design technological support.

In this section I will first review all of the claims I assert in the dissertation. I will then discuss contributions, and then future research directions. I will close with some final thoughts. All studies unless otherwise noted are conducted in the context of the class on biologically inspired design at Georgia Institute of Technology, as detailed in Chapter 4 of this dissertation. All claims made are with respect to the population and context of that class, which, for brevity, I will not articulate separately for each claim. I will conclude the claims section with a discussion on generalizing these claims and the implications outside this population and context will follow.

8.1 Claims

In this section I will review the research problem and hypothesis explored in each study, and I will reassert the resulting claims. As before, I will group the work into an exploratory phase, the content account, the process account, and tool implementation studies.

8.1.1 Exploratory Studies

While observational studies do not begin with a formal hypothesis, I view these observations from the perspective of cognitive science, with a bias toward processes of analogical design. In the observational studies I show that in biologically inspired design, the biological source of analogy plays a key role in the formulation and development of the design problem. I call this process Analogical Problem Evolution, (APE). The process is supported by two key process findings. First a problem may evolve through solution-based design, in which a problem is defined in response to a biological system. For example, the problem of a bullet-proof vest is defined in terms of abalone armor, and its ability to withstand impact. Second, a problem may evolve through compound analogy, in which a problem may be decomposed in the same way that a biological system decomposes that problem. For example, the problem of a robot moving stealthily underwater may be decomposed as two sub-problems, one in which the robot is moving quickly and one in which is moving slowly. This mirrors the way in which copepods use two different modes of stealthy movement, depending on their need for speed¹⁷.

I support this finding with two exploratory studies conducted in class. In the first study I explore my first hypothesis.

¹⁷ The copepod equivalent of a scene from Top Gun?

8.1.1.1 *Hypothesis E.1*

The introduction of biological analogues to student designers will yield greater range of concepts in a functional decomposition of a design problem, than a decomposition without biological analogue prompts.

In this study I examine a collective group problem decomposition activity. I show that student designers, when exposed to biological solutions expand their conceptualization of the problem formulation, in this case adding **50% more new functions** across all major branches of the decomposition and **at multiple levels of abstraction**. Figure 9.1 shows the incremental addition of new functional concepts, and the biological sources to which they can be attributed. While this exercise was facilitated such that the problem decomposition was constrained to function concepts only, the exercise revealed that student designers think about problems across many conceptual dimensions, for example structural and environmental factors. This led to a second experiment, as follows.

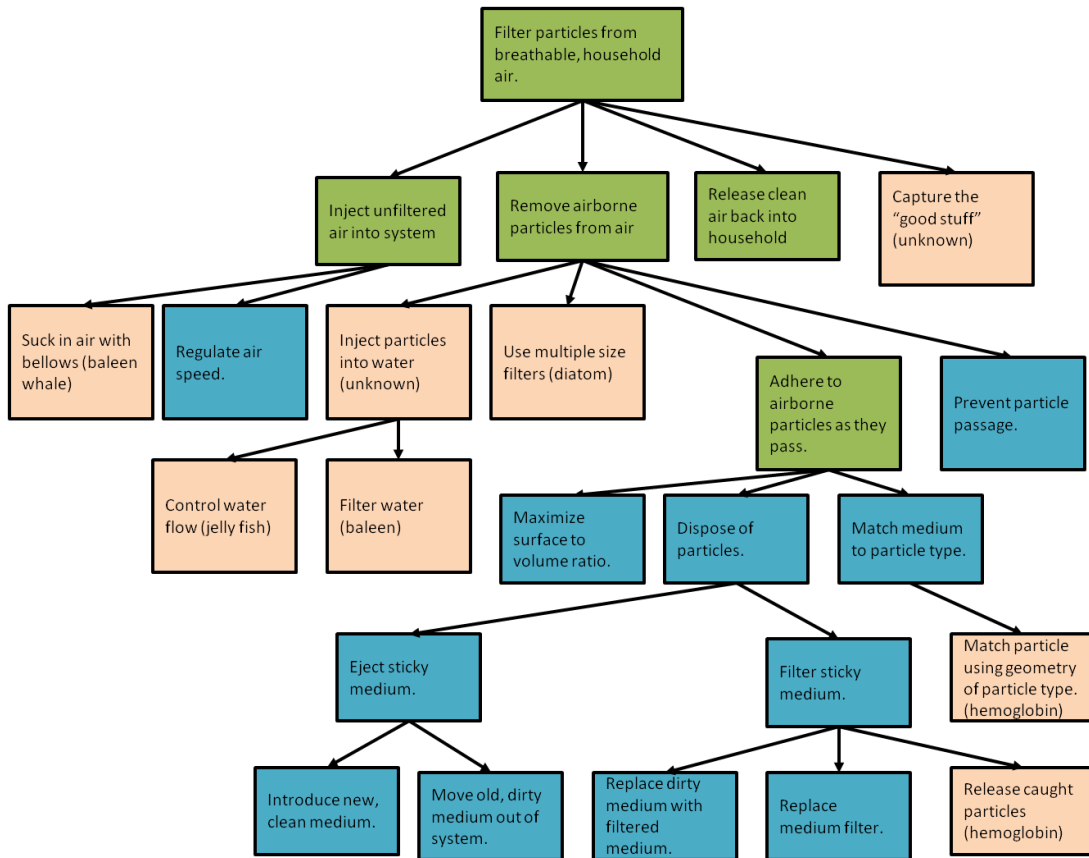


Figure 8-1. The final problem decomposition of a filtration design problem created during an in-class exercise. Green boxes represent the initial (given) decomposition, blue represent the decomposition after a single iteration, pink represent the decomposition after students were provided with biological analogue systems.

8.1.1.2 Hypothesis E.2

Student problem decompositions will follow a mixed conceptual decomposition strategy.

In this experiment, I analyzed the functional decompositions students generated with respect to a design problem that a biological system solved, for example the problem of attracting pollinating insects to a flower. While instructions and examples were *functional only*, students generated problem descriptions using a mixture of conceptual types. This mixture included functions (40.6%), functional refinements (5.4%), structures

(26.9%), external factors (5.2%), solutions (18.2%), and causal behaviors (3.8%). Coding of decompositions revealed patterns in problem decompositions with decomposition dominated by (a) functions, (b) function-structure, (c) solutions, or (d) mixed formulations. This experiment not only revealed that problem formulations are conceptually diverse, and that existing solutions accounted for nearly 20% of overall concepts. Furthermore, several patterns were revealed, including patterns that used existing solutions to frame decompositions.

From the collection of these findings: solution-based design, compound analogy, biologically influenced functional decomposition, and solution-oriented patterns of problem decomposition, I make the following claim:

8.1.2 Claim 1

Analogical problem evolution (APE), the incremental alteration of a design problem in response to a biological system, is a fundamental characteristic of the process of biologically inspired design.

Prior to these studies design practitioners and researchers viewed biological systems as inspiration for solution generation. While the role of biological system in problem evolution was occasionally noted, it was dismissed as a secondary effect. These studies elevate the process of problem formulation and evolution to a first class object of study, equivalent to solution generation in terms of richness, complexity and importance. That a biological source analogue may influence problem formulation opens up broad, new avenues for exploring how biology may be exploited (in the positive sense of the word) to increase innovation and creativity.

My studies also show that problem formulation in design is a cognitively challenging

task. A poor problem formulation may result in searching for solutions down many blind alleys, wasting valuable time and resources. An understanding of problem formulation and evolution, in addition to enhancing creativity and innovation may also provide basic scaffolding to assist with more routine problem formulation and evolution challenges.

While some existing theories address aspects of the phenomenon, for example analogical design *or* problem specification, no current theory current exists at the intersection of problem formulation, analogical design and problem evolution. In order to understand and leverage this phenomenon, in this dissertation I create an account of design problem formulation and evolution that could explain the phenomenon in terms of why, what, how and when problem evolution occurs in the context of biologically inspired design. I break the development of this theory into two phases: a content development phase in which I attempt to understand *what* and *when*, and a process development phase where I build an account of *how* and *why*.

8.1.3 A Content Account of Analogical Problem Evolution

I begin the development of an account of APE with the development of a knowledge representation schema for design problem formulation and the relationship to existing solutions, which I call a content account. The content account provides an ontology, or schema, for representing a problem formulation. It describes the conceptual contents and the relationships among those concepts, at least at one level of abstraction. The ontology is developed with respect to the process of Analogical Problem Evolution, which circumscribes the domain of interest. The ontology focuses on describing those concepts related directly to problem specifications and existing solutions, while excluding other concepts, such as design processes or process control (e.g. concepts described in solution-

driven design process account), interdisciplinary communications issues (e.g. design team interactions), learning issues (e.g. biologist understanding of engineering concepts) or support environment considerations (e.g. search tools, reference gathering).

I use the ontologically grounded theory method to develop the problem description ontology. In ontologically grounded theory I begin with a seed ontology that is already grounded in the domain of study. Beginning with a seed ontology concedes at the outset that the “grounding” is biased, and provides traceable specifications for that bias. In order to identify an appropriate theory I hypothesize that a content account must support six key aspects of analogical problem.

8.1.3.1 *Hypothesis C.1*

SBF provides a partial content account of analogical design that may be used as a seed ontology to discover the underlying account of problem formulation and evolution in BID.

Using ontologically grounded theory, and a three phase approach, I analyze over 60 student problem formulations to derive an ontology for the representation of problem formulations and their relationship to existing solutions. I call the ontology, shown in Figure 8-2, Structure Representations for Biologically Inspired Design (SR.BID), and develop a detailed rubric for encoding text-based problem formulations (see Appendix F). Using two methods of coding, independent coding with inter-rater reliability checking, and using co-coding with intra-rater reliability checking, for the selected level of granularity I establish that SR.BID ontology *reliably* encodes greater than 97% of all problem formulation related concepts. From these encodings, I generate problem models (Figure 9.x) which may be analyzed and compared over time. For example, Table 8-1

shows the distribution of concept types over 31 encoded problems. This analysis again emphasizes the influence existing solutions exert on the problem formulation, accounting for 18.9% of overall concepts, and occurring at least once in all problem formulations.

Table 8-1 Non-weighted mean percentage, standard deviation and frequency of occurrence of each category.

	Mean	Standard Deviation	Frequency
Functions	25.1%	9.1%	97%
Solutions	18.9%	11.3%	100%
Operating Environments	26.9%	15.7%	97%
Performance Criteria	5.3%	5.5%	61%
Deficiencies/Benefits	4.3%	5.4%	52%
Constraints/Specifications	5.6%	8.5%	42%

This experiment supports the following claim:

8.1.4 Claim 2

I claim that SR.BID provides a comprehensive and reliable account for representing problem descriptions, biological analogues and the relationship between them.

While a number of problem formulation representations exist, from rich requirements specification documents to function-based problem formulations, no representations exist specifically for the context of biologically inspired design within which the processes of analogical search, evaluation and transfer are essential. The SR.BID problem representation, grounded in SBF and developed using a ground up, data-drive process provides an essential problem-solution bridge. For researchers, this bridge provides a

reliable, comprehensive tool for describing and analyzing observations of problem-evolution processes. For practitioners, the representation provides a tool for the specification of problems and solutions in such a way that it facilitates the essential processes of analogy. I provide evidence in support of both of these points in the next two sections.

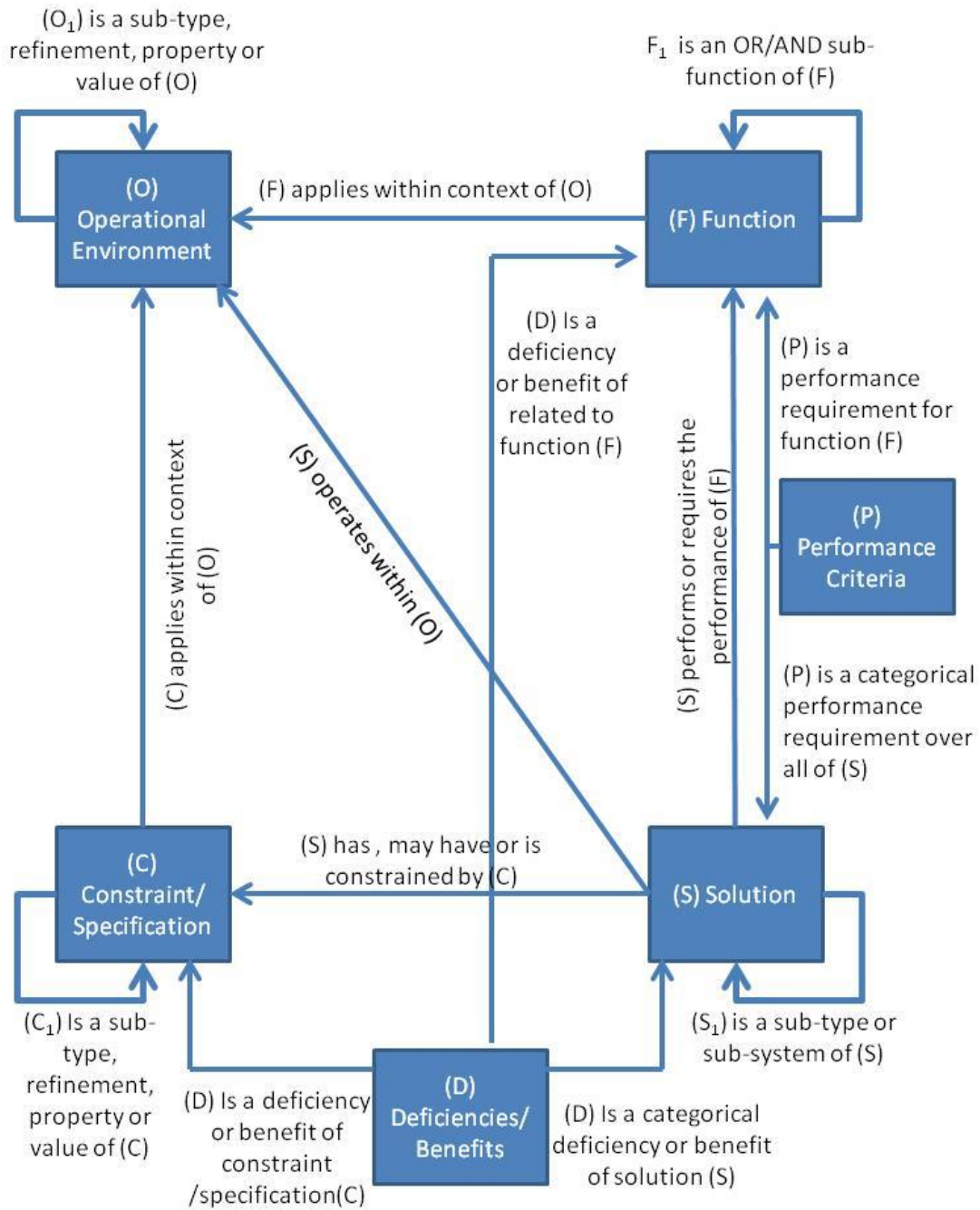


Figure 8-2. The primary conceptual categories and relationships of the SR.BID content account.

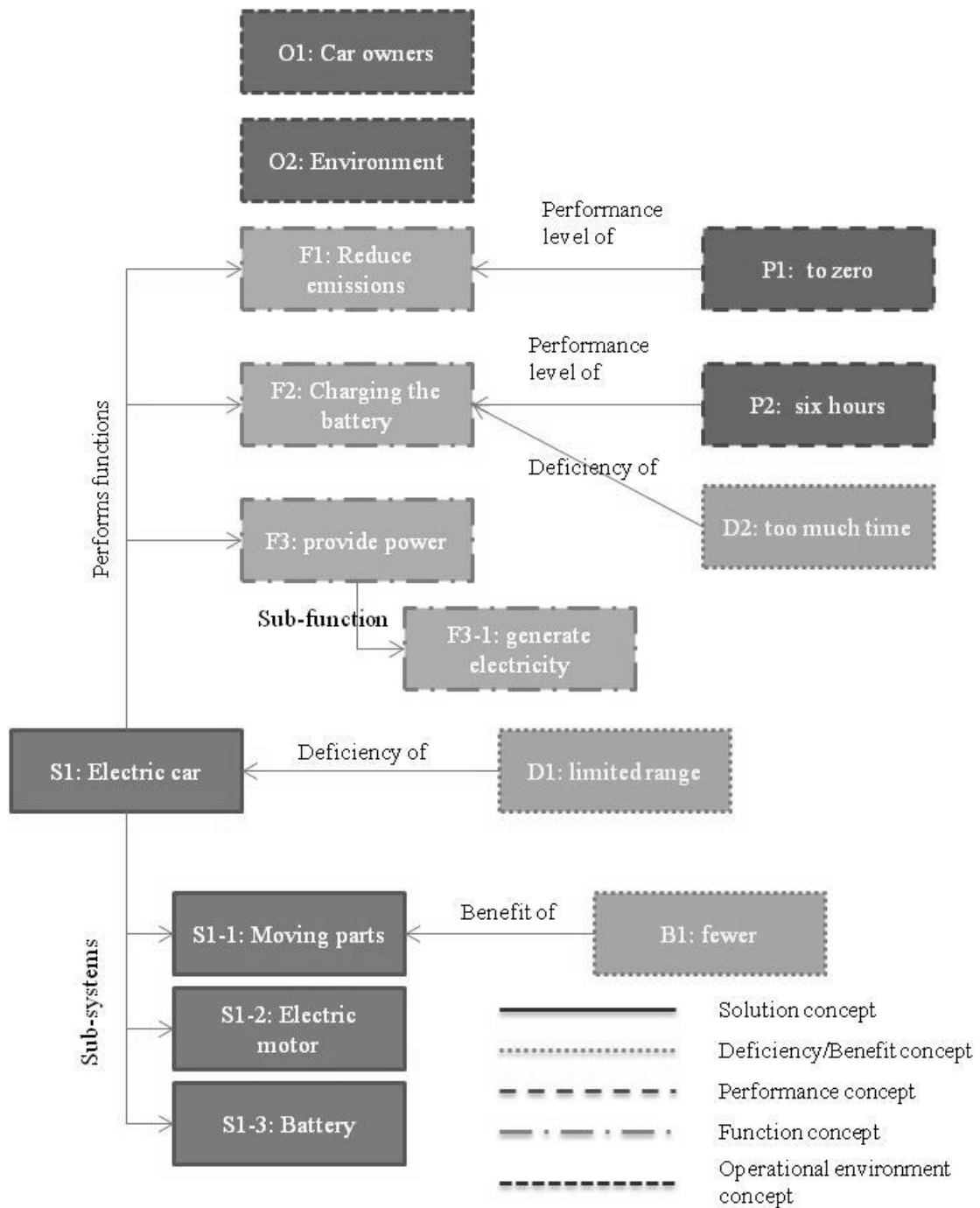


Figure 8-3. Sample SR.BID problem model.

8.1.5 A Process Account of Problem Evolution

My goal is to develop a rich process account of analogical problem evolution. To do so, I first desire a precise description of the process that details *what* content is transferred and *when* it is transferred. I believe that the SR.BID representation developed earlier will provide a useful method for developing such a description, as articulated in the following hypothesis:

8.1.5.1 Hypothesis P.1

An encoding of design problem formulations in terms of SR.BID will demonstrate the content transferred between biological solutions and design problems in the process of BID.

I use the SR.BID representation to encode problem formulations from a single design trajectory observed as a series of snap-shots over time. These encodings are abstracted into problem models, and compared over time using differential analysis, which identifies conceptual additions, deletions, and changes to relationships. It also identifies when concepts are dropped, but reappear in later problem formulations. Through differential analysis and SR.BID, I can track *what* and *when* concepts appear, and distinguish through relationship encodings which concepts are related to biological solutions. This provides a more direct “line-of-sight” on the process of analogical problem evolution. Table 9.x provides a sample analysis at the level of function.

Table 8-2. . Function concepts, by reference to man-made, biological or unknow sources.

	Week 3	Week 6	Week 9	Week 12
Man-made	18	4	10	4
Biological	0	12	6	6
No reference	11	6	12	3
Total	29	22	28	13

This table shows a summary of function concepts from problem formulations collected at 3 week intervals, and their relationship with either “man-made” or “biological” systems, or “no reference” when there is no explicitly stated relationship to an existing solution). Of the 29 total initial function concepts included in the problem formulation, 18 were related to man-made concepts, while for 11 concepts, no relationship to man-made or biological sources was specified in the document. In week 6 we see 12 concepts related to biological solutions, reduced to 6 in week 9 and 6 in week 12. This suggests that functional specifications of a problem specification are initially defined while considering how existing solutions address them; three weeks later this shifts dramatically to consideration of how biological solutions address the problem. In the final problem formulation, we see a blend, where 6 of 13 functions are related to biological solutions, while 4 are related man-made.

The following summarizes the findings from this study:

- In the context of this biologically inspired design class, the problem space explored is roughly 5 times larger with respect to functions than the problem space addressed in the final design. Over 80% of the functions explored are discarded by the final design phase.
- Either existing man-made, existing biological solutions or both are cited with respect to the formulation of the problem; this occurs in every problem formulation thus far observed.
- Concepts from biological solutions are not used in the *initial* problem formulations; rather existing man-made solutions provide the basis for many of

the concepts in the initial formulation. Concepts from biological solutions are integrated only in later stages.

- Certain conceptual categories in problem formulation are transferred more frequently from biological analogues than others. For example, while 38% of functional concepts appear to be transferred from biological analogues, only 20% of performance criteria are found in common, only 17% of operational environments, and no specifications/constraints appear in common.

From these findings, and based on my experience with biologically inspired design, I conjecture a process model, which I call PE.BID for problem evolution in the context of biologically inspired design, as represented in the following hypothesis:

8.1.5.2 Hypothesis P.2

Analogical problem evolution can be explained in terms of the PE.BID model of process evolution.

Figure 9.x provides a high-level diagram and major components of the model, which are meant to explain at one level of abstraction *why* and *how* analogical problem evolution occurs. The process model at this level provides a partitioning of the model into sub-components for investigation at a lower level of abstraction. In this dissertation I provide experimental evidence and claims for some of these sub-components, while providing only specifications and hypothesis for others; for this section I will provide only an overview for each component. I refer the reader to Chapter 7 for complete detail of the underlying specifications and experimental evidence in support of each component.

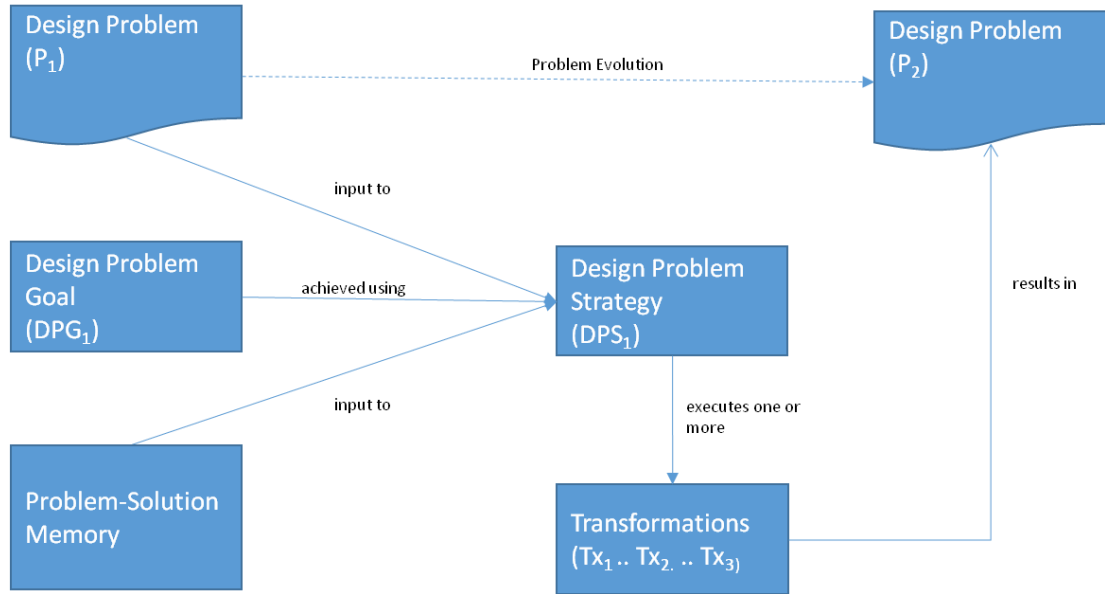


Figure 8-4. Graphical representation of the PE.BID theory of problem evolution in biologically inspired design

8.1.5.3 *Design problem*

I assume the design problem formulation may be expressed in terms of SR.BID, as detailed in the previous chapter.

8.1.5.4 *Designer problem goals (DPGs)*

I assume that design is a complex problem solving activity. According to (Funke 2001, 2003), complex problems exhibit five characteristics that simple problems do not. Funke proposes problem solver goals which address each of these characteristics. I hypothesize that designers generate similar problem goals to resolve the difficulties that arise from these characteristics in design, as shown in Table 8-3. Designer problem goals connect to designer motivation, and provide at least a partial answer for “why” problem evolution occurs.

Table 8-3. The five characteristics of complex problems and associated goals.

Characteristic	Description (verbatim, Funke, 2012)	Goal
Intransparency	Intransparency concerns the variables involved and the definition of the goal. In an intransparent situation, not all required information about variables and possible goals are given.	Intransparency requires from the problem solver the active acquisition of information.
Complexity	Complexity is defined based on the number of variables (concepts) in the given system.	Complexity demands from the problem solver a simplification through reduction.
Connectivity	It is not the pure number of variables that is decisive for the workload on the problem-solving person, but the connectivity between these. Assuming that in a system of 100 variables every variable is connected to only exactly one other, the connectivity is lower than in a system in which all variables are connected to each other.	For making mutual dependencies understandable, a model of the connectivity is required from the problem solver.
Dynamics	This feature explains the fact that interventions into a complex, networked system might activate processes whose impact was possibly not intended. It signifies that in a lot of cases the problem does not wait for the problem-solving person and his/her decisions, but the situation changes itself over time.	Dynamics requires from the problem solver the consideration of the factor "time."
Polytely	Usually there is more than one goal in a complex situation that has to be considered. These goals may be in conflict.	Conflicts due to antagonistic goals require the forming of compromises and the definition of priorities.

8.1.5.5 Designer problem strategies (DPSs)

Table 8-4 provides a small conjectured set of design problem strategies relative to addressing the first two designer goals (intransparency and complexity). The design problem strategy provides a high level process control for problem evolution, and associated knowledge requirements.

Table 8-4. Hypothesized design problem strategies associated with designer problem goals

Goal	Strategy	Description
Active acquisition of information	Breadth-first addition	Loosely related concepts are added in a breadth-first fashion, expanding the design space.
	Depth-first addition	Sub-concepts are added to existing concepts, generating conceptual depth for a particular concept.
	Relationship addition	Relationships are established between concepts
Simplification through reduction	Elimination	Concepts are removed from consideration in the problem space
	Decomposition	Concepts are divided into sub-concepts that can be considered independently; an interface may be necessary.
	Abstracting	One or more concepts of the same type at the same level of abstraction give rise to a concept at a higher level of abstraction.

Assuming the problem goal is known, each design strategy is comprised of several components as follows:

- (a) One or more *concept targets* that serve as both the index to required knowledge, and the focal point of the transformations.
- (b) One or more *possible transformations*, that determine what kind of knowledge is required and how that knowledge is applied to achieve the strategy,
- (c) *Knowledge requirements*, determined by the transformations to be applied
- (d) An initial *problem state*, and
- (e) A resultant *problem state*.

8.1.5.6 Problem transformations.

Problem transformations specify the low-level operations that may be used by a strategy to change the problem. Derived from observations of problem formulations, table 8-5 provides a list of problem transformations associated with conceptual

addition. A complete list of transformations with detailed descriptions may be found in Appendix H.

Table 8-5. Partial set of problem transformation primitives

Type	Sub-Type	Tertiary Type	Start State	End State
Addition	Refining		A	$A \rightarrow (r)B$
	Associating		A	$A \rightarrow (a)B$
	Abstraction shifting	Shifting-up (zooming out)	A(1-1)	$A(1) \rightarrow A(1-1)$
		Shifting-down (zooming in)	A(1)	$A(1) \rightarrow A(1-1)$
	Induced abstraction		A(1-1), A(1-2)	$A(1) \rightarrow (A(1-1), A(1-2))$
	Decomposing	Conjunctive	A(1)	$A(1) \rightarrow (A(1-1) \text{ AND } A(1-2))$
		Disjunctive	A(1)	$A(1) \rightarrow (A(1-1) \text{ OR } A(1-2))$
	Disconnected		--	A

A series of problem transformations may be used to explain more complex differences between problem models. For example, table 8-6 provides a list of 11 transformations to describe the change from problem formulation 2 (figure 8-5) to problem formulation 3 (figure 8-6). This complete list of transformations is supported by the derivation of the transformations from observed data, and the ability of the transformations to explain observed changes in problem formulations.

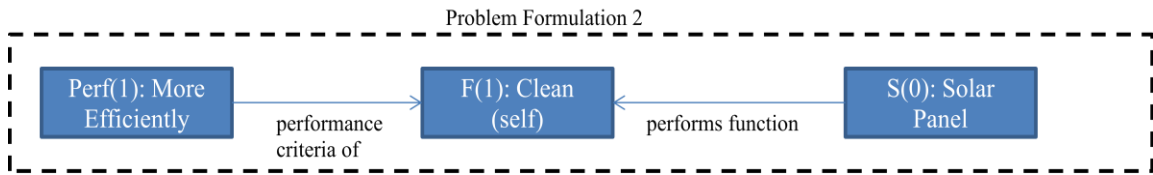


Figure 8-5. Partial problem model from problem formulation 2.

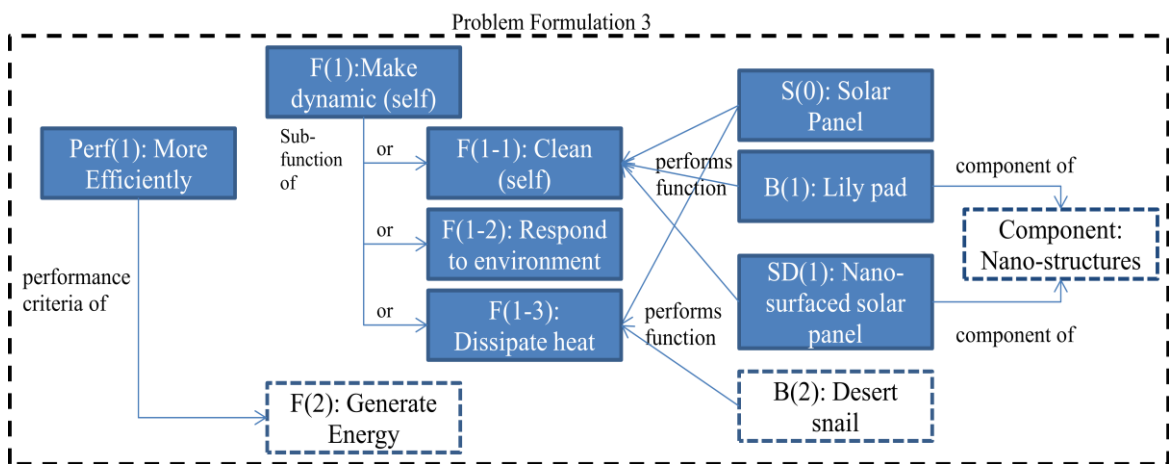


Figure 8-6. Partial problem model from problem formulation 3.

Table 8-6. Transformations required to move from problem formulation 2 to problem formulation 3.

Formulation 3 Transformations			
Step	Concept(s)	Transformation	Related Concepts
1	B(1)	Associative addition	F(1-1)
2	B(2)	(second order)	--
3	F(1-2)	Associative addition	B(2)
4	F(1-3)	Disconnected addition	--
5	F(1)	Induced abstraction	F(1-1, 1-2, 1-3)
6	F(4)	(second order)	--
7	Perf(1)	Connection switch	F(4)
8	Spec(1)	(second order)	--
9	SD(1)	Composition	S(0), Spec(1), F(1-1)
10	F(1-1), B(1), Spec(1), SD(1), S(0)	Partition (1)	--
11	F(2), B(2), S(0)	Partition (2)	--

8.1.5.7 Problem-solution memory

Because SR.BID, a ontology for problem representation, was grounded in SBF, a model for solution representation, there exists a natural union of the two ontologies that may be used to represent a memory that uses common underlying concepts, such as function, structure, etc. Table 8-7 shows the relationship between key SBF solution concepts and the corresponding problem concept. I provide evidence for this shared memory construct through the instantiation of a database used for the SR.BID Web Application, explained in the next section and detailed as an entity-relationship diagram in Appendix K.

Table 8-7 Problem-solution model relationships.

Solution Concept	Problem Concept	Relationship
Structure	Constraints/Specifications	The problem model either prescribes specific components and relationships (specifications) for the structure, or a set of parameters which the structures must meet (constraints).
Function	Function	The problem model provides a set of functions the solution concept must perform
Behavior	Performance Criteria	The performance criteria aspect of the model provides a set of criteria against which the behavior of the solution can be measured
External Input	Operational environment	The operational environment aspect of the problem model provides the set of potential external influences which may act on the system.
Structure, Behavior	Benefits/Deficiencies	This aspect of the problem model serves to draw specific attention to aspects of existing solutions, (usually behaviors or structures) that should be included (benefits) or improved upon (deficiencies).

While the entire process model is not supported, evidence exists in support of certain components of the model, in particular for the problem representation, for problem-solution memory organization, and for the low-level transformations. From this evidence I make the following claim:

8.1.6 Claim 3

The PE.BID transformation set provides a minimum set of primitive transformations for explaining incremental problem state transformations.

The components of the PE.BID process have rich implications for cognitive science, AI and BID. The SR.BID problem representation component provides both a novel analytical tool for cognitive science research, and a means for supporting analogy-related tasks in BID practice. The problem-solution memory as an extension of SR.BID, provides both a novel interpretation of problem-solution memory organization for AI systems building, and provides a bridge between biological solutions and problem formulation. Since all problem formulation studies in this dissertation indicate rich connectivity between problems and solutions, such a “problem-solution bridge” in memory must play a critical role in the underlying process of problem formulation and evolution. As process and representation are intrinsically linked, future studies of the processes will necessarily depend on connected problem-solution models of memory. Furthermore, the problem-solution bridge is useful for BID practitioners for the process of both search and analogical evaluation. Finally, the documented list of transformations provides the fundamental operations at a level of specification at which computational systems may begin to automatically transform problems. This will certainly have implications on AI for engineering and design.

I mentioned several times the implications of SR.BID and PE.BID on the processes of analogy in the context of BID. My final experiments provide support for those implications, and end with a claim about the utility of SR.BID in the practice of BID.

8.1.7 The Four-box Method of Problem Formulation

In order to demonstrate the utility of SR.BID in the practice of BID, I deployed several tools in the context of the BID class at Georgia Institute of Technology. The main tool is the four-box method of problem specification, which is based on the categories identified in the SR.BID ontology. The first question with respect to the tool is the degree to which it can be used accurately by members of the class to specify a design problem, leading to the following hypothesis:

8.1.7.1 Hypothesis I.1

The four-box method can be used accurately by all students to represent design problems in BID.

In order to test this hypothesis, I deploy the tool in class in both 2011 and 2012 with a brief one-class period training session. Students are asked as part of a homework assignment to use the tool to represent a problem specification. I measure the degree to which students are able to accurately classify problem concepts in terms of the four major categories in the four-box specification (function, operational environment, performance criteria, and specification). In this study across two years, 75% of students are show to use the tool with greater than 80% accuracy, with no significant statistical variation by gender, major, year of deployment, or method of representation (4-box/bullet points or narrative/paragraph format). Figure 8-7 provides a histogram of overall accuracy for all students in 2011 and 2012. This suggests that the four-box method *can* be used equally well by all student designers, with a high degree of accuracy.

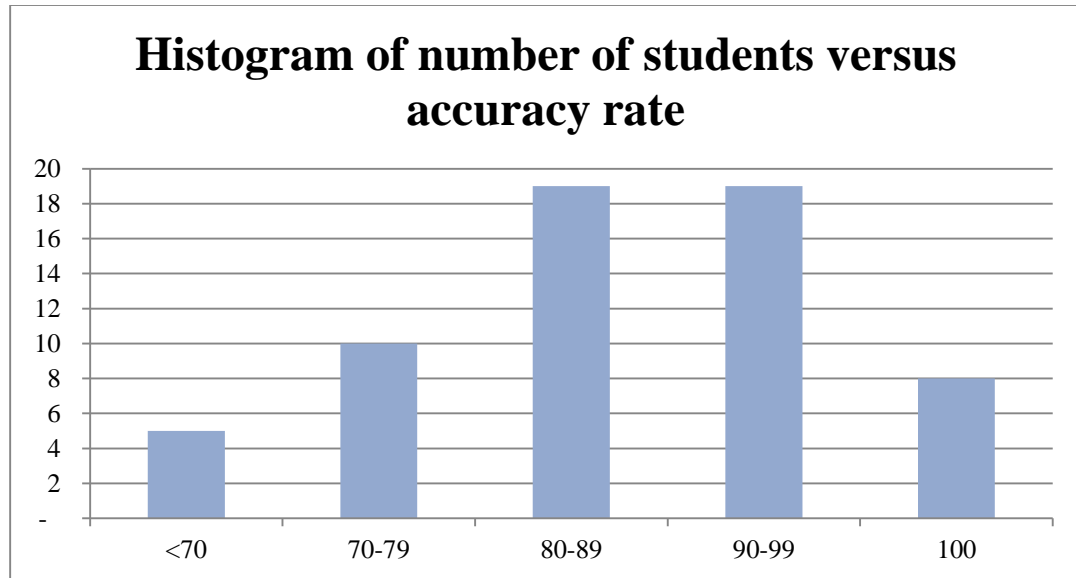


Figure 8-7. Histogram of accuracy of use of four-box method

Figure 8-8 shows that while students are able to use the method accurately, there is some variation in the accuracy with which certain concepts are able to be accurately categorized. In particular students appear to be able to accurately categorize environmental concepts, but are less able to categorize specifications and constraints.

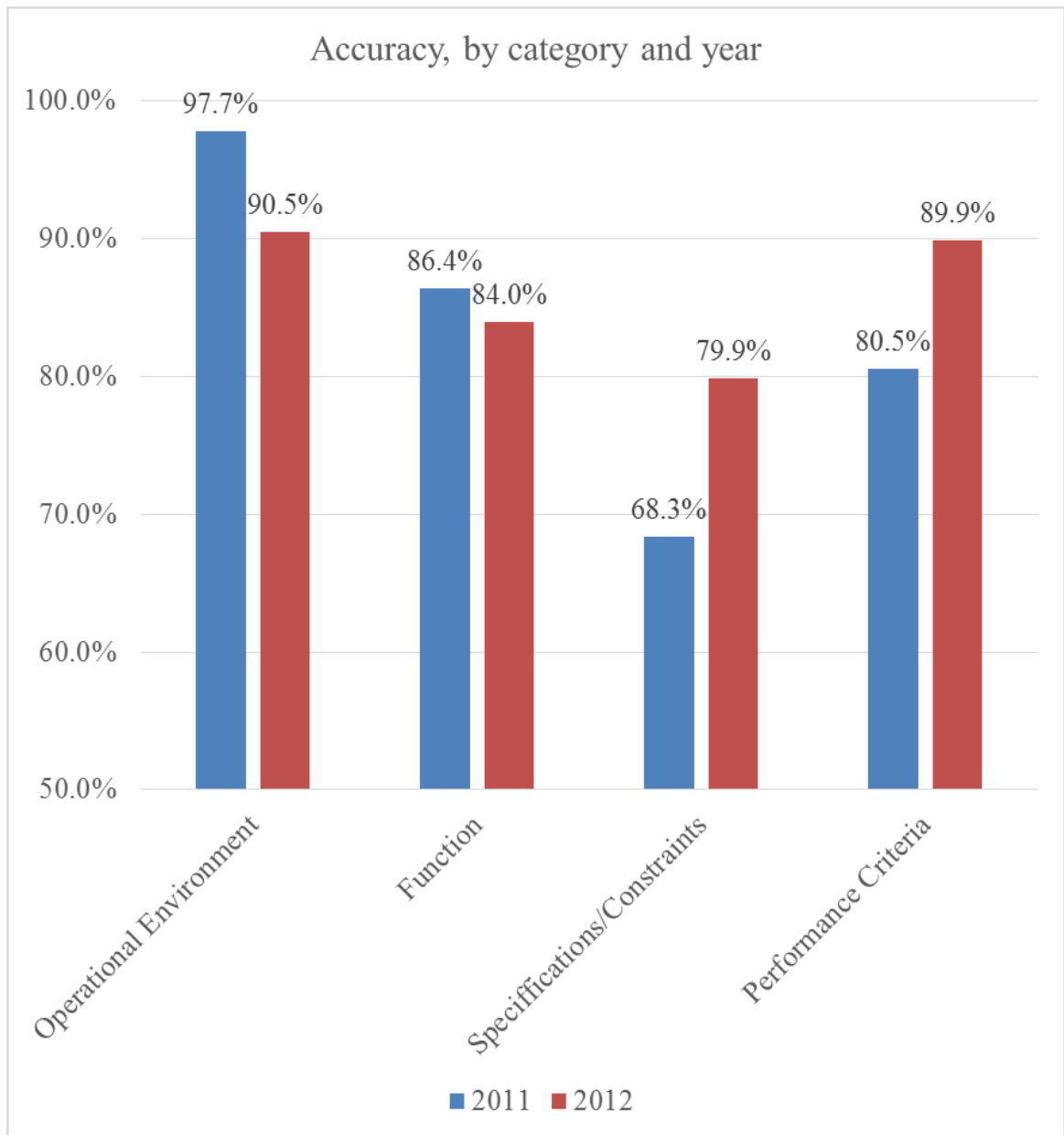


Figure 8-8. Differences in accuracy of 4-box method, by category, by year.

The next hypothesis and experiment explores how SR.BID and associated tools are used in class, and what value they provide.

8.1.7.2 Hypothesis I.2

SR.BID will aid students in defining and communicating design problems and biological systems, and help in understanding and communicating their analogies.

In 2012, students used the four-box method to define design problems, and use a modified form of the four-box method, called T-charts, to evaluate analogies charts. In T-charts, design problems and biological solutions are evaluated against the four main categories of the four-box method (see section 4.3.3.4 for a complete description of T-charts). At the end of the semester, as part of a take home exam students were required to answer reflective questions over the tools and techniques they learned during the course. Many of these comments were directed at the four-box method, the T-chart, and SR.BID.

Using a qualitative assessment of these reflections, I determine student use and perceptions of the utility of SR.BID, the four-box method and T-charts. During the qualitative assessment, student reflections are parsed into ideas (usually one to three sentences in length), and categorized according to general patterns as they present themselves (that is, the general patterns are not prescribed beforehand). Table 8-8 shows student comments with respect to the four-box method. On the left hand side of the table I report positive comments, grouped by category, with the number to the right of the comment representing the number of times the comment appeared. The right hand side shows critical or negative comments. From this synthesis I draw two conclusions about the four-box method.

First, I conclude that first four of the comments on the positive side provide ***strong evidence for the value of the four-box method of problem specification***: in particular that, as intended, ***it provides students with a greater capability to define/specify/clarify,***

breakdown, organize, and otherwise grapple with complex design problems. Second, since positive comments outweigh negative comments to a 3-to-1 ratio, I conclude that students found the four-box method generally helpful. The critical comments do suggest room for improvement with respect to training. The critical comments also expose weaknesses in the four-box method, with respect to limiting “creativity” and limiting design thinking to a single environment. One could interpret this as a reaction to what is otherwise perceived as a strength of the method; it is intended to help students focus and to narrow the design scope to something achievable in the time they have available.

Table 8-8. Comments, count by category for the four-box method

Positive Comments	29	Negative Comments	10
Define/specify/clarify problem	8	Decrease/Limit creativity	3
Breakdown, problem	6	Limited to a single environment	2
Focus	2	Confusing, categorizing concepts	1
Organize data/knowledge/problem	2	Confusing, redundant	1
Search, aid	2	Confusing, specification v performance criteria	1
Understand, system	2	Difficult to learn, to use initially, different at first	1
Analogy, matching	1	Increased workload	1
Direct inquiry	1		
Easier than another system	1		
Evaluate, problem	1		
Understanding, SR.BID	1		
Useful, operational environment	1		
Visualization	1		

Using the same method as above, I analyze student reflections on the T-chart method of analogical evaluation, and on the SR.BID representation. Although a complete analysis of the T-chart method is out of scope for this dissertation, the amount and type of comments (see section 7.4.3 for details) collected in the reflections speak strongly to the

value of the SR.BID representation schema when it is contextualized in a tool and a design task. The number of positive comments outweighs the negative comments for T-charts by a 2-to-1 ratio. What is particularly encouraging is that *30 positive comments (roughly 40%) are directed at the core value proposition of the tool: the use of SR.BID for analogical identification, selection and evaluation.* This further supports the hypothesis that the four-box method provides perceived value.

8.1.8 Claim 4

The four-box method derived from SR.BID can be successfully used by all students in the BID classroom for accurately representing problems in the context of BID.

When introducing a new tool to a population, that tool must always weigh the cost and time associated with implementation and learning, versus the benefit the tool will provide. In the context of the biologically inspired design class at Georgia Tech, I show that not only can students accurately use the four-box method with minimal training, but they also find the method valuable for at least two key tasks: problem formulation and analogical evaluation. The value and accuracy data from this study are necessary preconditions for my final study, in which I use SR.BID and the four-box method to create a scalable, web-based platform. This application is capable of supporting both design problem definition, biological system description, and biological system search using the problem-solution memory schema discussed in the PE.BID model.

8.1.9 The SR.BID Web Applications

I extend the value of the SR.BID, PE.BID and the four-box method through a web-based application, creatively named the SR.BID Web Application. The Web Application uses the underlying problem-solution memory discussed in PE.BID to enable designers to

specify problems and biological systems both in terms of the four-box model. In so doing, the system enables designers to specify a design problem and then to search a database of biological solutions using the same terms. This study represents a proof-of-concept system intended to demonstrate that students are able to quickly and easily add both problem specifications and biological systems into a shared memory infrastructure. In this study I test the following hypothesis:

8.1.9.1 *Hypothesis I.3*

The SR.BID for Web Application can be used to support designers for problem definition, biological analogy building, and analogical search and evaluation, over a distributed team-based platform in the context of BID, with minimal additional investment over current assignment workload.

In order to test this hypothesis, I implement SR.BID in the form of a web-based application for use in the last few weeks of the 2012 biologically inspired design class at Georgia Tech. The system is designed to be used by design teams to capture design problem descriptions and biological system descriptions, along with supporting documents and images. Descriptions are available in text and by using the four-box method. The application is designed to capture all of the project information required in the classroom in a sharable, team based virtual work environment.

A brief training is held in class, and students are asked to use the system as they see fit. There are no assignments or grades tied to the application; student use is entirely voluntary, outside of class at their own discretion. Students create their own login ID, which is used to track transactions completed on the system, for example adding problem descriptions or new biological systems. Projects may be viewed and edited by multiple

FOUR BOX DESCRIPTION

last update at: Nov 11, 2012 10:32EST Michael Helms

Operational Environment	Operational Env. Type	Options
Mouth, teeth, tongue, gums	Location	<input type="button" value="Delete"/>
Saliva, toothpaste, bacteria, dental plaque	Condition	<input type="button" value="Delete"/>
Adults, Children, Dentists & other medical profe	User	<input type="button" value="Delete"/>
<input type="button" value="Add Operating Environment"/>		

Function	Engineering Function Type	Options
Safe effective removal of dental biofilm	Remove/Block	<input type="button" value="Delete"/>
Dislodge plaque and other food particles	Remove/Block	<input type="button" value="Delete"/>
Prevent periodontal inflammation	Protect	<input type="button" value="Delete"/>
Prevent oral irritation and ulceration	Protect	<input type="button" value="Delete"/>
Deliver agents to the tooth surface	Provision/Supply	<input type="button" value="Delete"/>
Stimulate gum growth	Support	<input type="button" value="Delete"/>
<input type="button" value="Add Function"/>		

Figure 8-9. Partial screen shot of four-box entry screen.

users, determined by the project owners. Figure 8-9 provides an example screen shot for how a problem descriptions may be entered into the system in four-box format.

I allow the system to run for a period of weeks, and then collect a snap-shot of the database at the completion of class. All entries in the database are time stamped and associated with a user's login ID; this enables me to track when and how long users took to accomplish particular tasks in the system. Seven students generated login information for the system. Of those, two students engaged with the system and entered project descriptions, biological system descriptions or both. Analysis of the transaction logs shows that:

The first user, used the system on three separate occasions 11/8, 11/11 and 11/19.

Initially, the user created a new project with a short description, and added two references, a design report and a materials assessment. In total, they used the system for approximately 15 minutes over three sessions on different days, to create a new project, add three documents to share on the project, add one biological system including a detailed mechanistic account of a function, and add one reference paper for that system.

The second user (#7) focused more on the four-box problem description for their project. They used the system over two separate sessions on the same day, 11/9. They added a new project with a 400+ word description with 10 supporting documents, all related to the nature of the design problem (cleaning teeth). The user next added a complete four-box model of the problem, and added images of four hand-drawn design sketches, to the project. In total, this user spent approximately 42 minutes building a complete problem representation, including a 400+ word description, a complete four-box model, 10 reference documents and 4 images of design sketches.

These two use-cases show that

- The initial creation of design project descriptions including free-text, document and image support can be accomplished in between 10 and 30 minutes, depending on the level of detail and amount of supporting documentation provided.
- One user shows that a biological system can be added to the database with a similar level of investment
- The creation of a rich, complete four-box model for problem description can be accomplished in approximately 15 minutes.
- In less than one hour, a student designer can document a complete design

problem description using the four-box method, augmented with free-text, image, and documentation support.

This leads me to make the following, final claim.

8.1.10 Claim 5

I claim that viewed as an underlying scaffold for both tools and technology, SR.BID and the SR.BID Web Application could potentially be used for low cost, massive distributed collection of design problem and biological system information.

I show from studies I.1 and I.2 that SR.BID is both easily used and is perceived as valuable for the tasks of problem formulation and analogy evaluation. Moreover, student designers can use it accurately, with minimal training, with no difference between demographics within the class. This suggests SR.BID may be useful to a broad audience. Furthermore, the SR.BID Web Application shows that for very little incremental cost, rich structured representations may be annotated to unstructured descriptions; that is, text and image descriptions are easily and accurately annotated with four-box descriptions.

This is implemented within a database that uses shared problem-solution concepts to generate descriptions. If one could get many designers engaged in such activity, once sufficient critical mass is achieved with respect to the database size, it is easily extended to enable automatic retrieval of relevant biological systems for any given four-box problem specification. Moreover, it is easily extended to allow solution-to-problem search (e.g. solution-based design), and problem-to-problem search.

8.1.11 Generalization of Claims

In this dissertation I assert claims that are all circumscribed within the context of an

undergraduate classroom on biologically inspired design. While I make no formal claims beyond this context, I would like share some additional thoughts about the circumstances of the study and how they may compare with other design environments. I leave it to the reader to make additional inferences.

8.1.11.1 *Range of design problems*

In the context of the class, student designers were generally free to select whatever design problems they see fit; exceptions were made over two years in which the design problem had to conform to either being “water crisis” or “sustainable housing” oriented, although students had a great deal of latitude within those domains. The range of problems students explored was incredibly diverse. The following is a list of final projects from 2006-2007:

- Bullet proof vests
- Traffic congestion
- Air filter
- Thermo-regulating clothing
- Hip replacements
- Camouflaged surfboards
- Bomb detection
- Cell phone casings
- Color changing cars
- Bio-film resistant catheters
- Better rock-climbing shoes
- Collision detection

- Oil-spill clean up
- Impact resistant helmets
- Stealthy underwater robots
- Improved mine shaft supports

This list, which is fairly typical for the context, includes problems from mechanical, chemical, biomedical, and informational domains, encompassing solution principles from statics, fluid dynamics, thermodynamics, signals, organic chemistry, and systems engineering. The representation schema I developed for representing problems and solutions was developed in a context with data from this complex range of problem types.

8.1.11.2 *Existing solutions*

The theory of Analogical Problem Evolution was developed in a context where biological analogies are constantly in use, and are obvious when made. *However*, the data show that student designers in this environment also look to existing man-made designs to inform their problems formulation. If biological analogies were not mandated as they are in the context of study, to what extent would existing man-made solutions continue to be used in the process of analogical problem evolution? To what extent do engineers in general look to solutions from other, non-biological fields to inform their problem thinking?

8.1.11.3 *Experts versus novices*

The context of study senior-level undergraduate students from a range majors, most of whom came from engineering, bioengineering, or biology. These student designers should be considered novice engineering designers, in that they have little real-world design experience (although there were certainly exceptions). To the extent that these

studies are all about and limited to novice designers.

With respect to biologically inspired design, however, who could be considered an expert? For all of the promise and potential of the field, there are very few commercially successful applications of biologically inspired designs, and none that I know of that employed a so-called systematic biologically inspired design process. Successes to date are largely a result of serendipity, with no known cases of serial-designers, that is, designers who have demonstrated expertise by generating multiple commercially successful bio-inspired designs.

Moreover, biologically inspired design is inherently multi-disciplinary, thus in order to achieve expertise in the field, an individual must achieve expertise in multiple fields. While certainly possible, this raises the threshold and expectation one must have to be viewed as an expert; or we must change our definition to think not of experts and novices, but of design teams with skills that lie across the continuum.

I do not intend to imply that design experts in their respective fields would practice biologically inspired design in the same way that student designers practice BID, nor that my theories are applicable in those contexts. Finding differences between practitioners with varied levels of expertise in BID, is perhaps more complex than elsewhere.

8.1.11.4 *Constrained versus unconstrained design problems*

In the context of study, students are given a large degree of latitude with respect to the design problems they choose to work on. Students change their design problems over time, sometimes radically. There are many engineering and design circumstances where such radical alterations to the design problems are not possible. In real-world design companies, existing processes, manufacturers, or customers and the constraints that are

imposed by them limit the amount of change allowed to a design problem. In the extreme case, design problems become routine; minor modifications to parameters of existing, know components. These studies do not take place in the context of routine design.

Consider, all formal design processes conceived of to date. They all have at least two things in common: there is always a design problem definition phase, and there is always room for iteration. That is, all design processes implicitly acknowledge that there exists room at the start of a project to define a design problem over some larger possible design space. Moreover, every modern process accounts to greater or lesser extent for iteration on the process of problem definition. While some designs are more constrained than others, there is nearly always at least some room with respect to design problem definition. The theories in the dissertation are theories of *how*, *why*, *what* and *when* problem evolution occurs; they are not theories of *how much* problems change.

8.2 Contributions

In this section I outline the contributions this dissertation makes to various fields of study and practice. I break the contributions into five categories: cognitive science, artificial intelligence, design science, biologically inspired design, and biologically inspired design pedagogy. I differentiate the last two, as their needs are different.

8.2.1 Cognitive Science

This dissertation uses the context of biologically inspired design as an environment in which to study the cognition of analogical design. In this context, I expose the phenomenon analogical problem evolution, in which an analogical source is used to inform the designer's personal interpretation of their design problem. Within the scope of

cognitive science research on analogical design, in exposing the phenomenon, *I elevate the study of the design problem to the same status as the study of the design solution.*

While some theories of problem definition have been espoused over time, no cognitive theory of design has explored problem inception, definition and change to the level and extent to which solution generation and change has been explored. I believe this is particularly meaningful for analogical design, as the definition of the design problem – long known to be dynamic -- plays such a fundamental role in all of the processes of analogical design.

In developing SR.BID, the content model of problem representation, I provide three very important contributions to cognitive science. The first is *the SR.BID representation itself*; which becomes a useful framework for the discussion and definition of design problems and the processes that operate over them. Because representation and processes are intrinsically bound by providing the problem representation, I am also in part defining all processes which operate over them. In addition to problem evolution, this encompasses, for solution generation and evaluation which depend (one assumes) on the way in which the problem is defined, as well as processes such as traditional analogical design.

The second contribution is the *relationship between the problem definition and the solution definition*. The relationship between problem and solution is inherent and exposed in the SR.BID representation because of its grounding in the SBF ontology. As a result, the representation neatly bridges the gap between the two, providing a common set of underlying structures upon which both may be constructed.

The third is the SR.BID representation, in conjunction with the rubric, as *an*

analytical tool. The representation enables the reliable and comprehensive encoding of problem formulations, such that aspects of problem formulations may be correlated with other factors. In the case of this dissertation, I correlate changes in problem formulation with exposure to biological analogues to infer when and what concepts are transferred. In other cases, one may wish to correlate measure of creativity, feasibility, cost, team behaviors, etc. with aspects of problem formulation.

These studies also show that all problem formulations across all problem types refer to existing solutions. This finding is evident in data sets explored in this dissertation, and emphasizes the dominant role existing solutions play in the formulation of design problems.

8.2.2 Artificial Intelligence

The content model of problem formulation is as much a contribution to artificial intelligence as it is to cognitive science. The SR.BID formulation is more than a “boxes and arrows” theory of problem representation; it is a computationally instantiated and useful representation. This provides an immediate benefit to AI theories of design, especially those theories that currently use impoverished problem formulations. In the SR.BID Web Application I show that SR.BID may be applied as an underlying memory structure for both problem and solution definition using a shared set of more primitive base concepts. This shared problem-solution memory informs (although I do not provide answers for) a number of questions in AI; for example, given a problem, how are analogical solutions retrieved (as in analogical design); or given a solution, how might problems formulated (as in solution-based design)?

Secondly, the model of problem evolution (PE.BID) provides a number of

specifications, grounded in data, that an AI system may use to inform the process of problem evolution. In particular, the rich system of transformations outlined provides operators at a level of detail that can be computationally implemented. The framework for strategies provides the knowledge components that will be needed, and the framework of goals provides a high-level motivation that may be linked back to user objectives. As mentioned already, SR.BID also provides an organization for memory, with implications for the implementation of processes.

Finally, knowledge engineering is a known problem in knowledge-based AI. One solution is to hire well versed, trained knowledge engineering experts – a practice that is both expensive and time consuming. Alternatively, the internet provides massive amounts of information that can be statistically analyzed, from which inferences can then be drawn. This is closer to the machine learning paradigm than knowledge-based AI, and while useful for question answering (mostly) it is a questionable long-term approach for human-level understanding. Group-sourcing is a third potential lever, however AI systems that require complex underlying models are often too fragile to suffer the inaccuracy that is inherent in group-sourcing practices. I show in this dissertation that the SR.BID tools I developed enables the collection of accurate, structured knowledge representations with low time over-head, and which student designers perceive as valuable. As any system that interfaces directly with human users in the real world must, SR.BID strikes a balance between the investment required of the user and the value returned. Such a system can be used to attain the critical mass of structured knowledge necessary to enable automated BID system supports. Once value is being delivered, the systems of representation can be incrementally made richer, delivering more automated

value as the richer representations enable more automated inferences.

8.2.3 Design Science

In much the same way as this dissertation contributes to Cognitive Science with a representation for analyzing problem formulation, the same argument provides value to the field of design scientists. A problem ontology and a comprehensive and reliable methodology for using it to describe and analyze problem formulations provides a scientific tool for the continued exploration of how problem formulation influences various aspects of design.

This dissertation also makes a strong case to design science for deeper investigation of the role existing solutions have on the formulation of design problems. This applies not only to biologically inspired design, but to all design. It is said that innovation is as much about finding good problems as it developing good solutions. Knowledge of existing solutions provides scaffolding for the rapid development of thinking about new problems; but does it also constrain thinking about potential new problems?

This dissertation outlines a number of processes of interest to the design science community. While there exist several “takes” on solution-based design, the process description I provide in this dissertation of solution-driven design is novel emphasizing to an extreme the influence a solution may have on the development of a problem, in this case, on the problem inception itself. The process reinforces the deep connection between existing solutions and problem formulation.

In a similar way, the process of analogical problem evolution which is the main driver of this dissertation provides a novel contribution to design science. While the phenomenon of problem evolution is well known, current theories of design problem

evolution focus on evolution with respect to a generated or conjectured solution, often framed in reaction to failure. The phenomenon of analogical problem evolution shows that problem evolution may also occurs as a reaction to exposure to solutions. This exposes to design scientists a new method for innovative design.

8.2.4 Biologically Inspired Design

This dissertation contributes to biologically inspired design in three ways. First it exposes an extremely valuable aspect of biologically inspired design that until now has not be recognized. The value of the biologically inspired design process has been ascribed to its (theoretical) propensity to generate novel solutions through the transfer of mechanism or form from the domain of biology to the domain of design. In this dissertation I demonstrate that at least some part of the power of the biologically inspired design process lies in the ability to inspire problem formulations. These problem formulations may, and often do, lead to engineered solutions that are wholly engineered from existing man-made principles; but are combined in a novel means because of the departure from traditional thinking about the problem itself. The recognition of analogical problem evolution as a first-class effect, which can now be recognized and sought, fundamentally alters the way BID practitioners view and value their process.

This dissertation also provides a valuable tool based on the SR.BID ontology that enables the practice of biologically inspired design. The four-box method provides a tool for analogy-enabling problem formulation; that is, it not only aids in the problem formulation task, but it does so in such a way as to facilitate the evaluation of analogies. No tool or formal method currently exists which considers problem formulation, solution

description and analogical evaluation using a common, comparable representation. With this one tool (and its extension, the T-chart) BID designers can easily categorize their problem, find and classify biological systems, and compare them directly for fitness across multiple dimensions.

Coupled with the four-box method, this dissertation also provides the SR.BID Web Application, a technology usable and useful to the BID community for developing structured representation of problems and solutions. I show that this technology can provide automated search, retrieval and evaluation of analogies at a very low incremental cost. As a result of the underlying PE.BID problem-solution memory model, The technology also enables two novel search methods: solution-to-problem search, in which a designer may discover new problems to work on, based on an interesting biological solution; and problem-to-problem search in which a designer may discover design problems similar to their own, which may lead to novel problem re-formulations, and possibly new solutions as well.

8.2.5 Biologically Inspired Design Pedagogy

I will close the contributions section with contributions to the field of biologically inspired design pedagogy. Many of the theories developed in this dissertation are already incorporated into the instruction of BID at Georgia Tech. The class was fundamentally restructured to incorporate an iteration of solution-based design, ensuring that students understood and could apply this novel design approach. Instructors directly applied and continue to use (as of Fall 2013) the solution-based design theory and process in their education practices.

Second, as with BID in general, as a result of this work instructors recognized that

problem definition played a fundamental role in the process of biologically inspired design, and furthermore that students struggled with it. As a result, students are now trained and provided with representational scaffolding (SR.BID and the four-box method) to support the process of problem definition.

Third, no formal method or scaffolding for analogical evaluation had been available until the T-chart method was developed (based on SR.BID) in the 2011. This method now provides students with a straightforward means to evaluate one or more analogical sources against their design problem specification. Both the four-box and T-chart methods are shown to be easy to use, with perceived value for the tasks for which they were designed. The four-box and T-chart methods continue to be used in the class, as of the writing of this dissertation (2013).

8.3 Future Research

This dissertation centered on the phenomenon of analogical problem evolution, and developed a number of tools and techniques for the exploration of design problems and the evolution of those problems over time. While this dissertation touched on many of these tools and techniques, each provides additional opportunity for study.

8.3.1 Design studies of analogical problem evolution

As a major driver of this research, I developed tools to explore the phenomenon of analogical problem evolution. The assignments and tools developed for the biologically inspired design class provide a rich set of tools for gathering text-based data in design scenarios where the researcher has some degree of control. SR.BID provides a robust system of encoding and analyzing this data. CBID and the Design and Intelligence Lab possesses data from 50+ designs conducted since 2009 in the BID@GATech class that

could be analyzed using SR.BID and the techniques developed in this dissertation. Analysis of this data could reveal additional patterns for when and what information is transferred in the APE process, and is required to begin making generalizations about APE even within the BID context. This form of analysis will also provide opportunities to deepen the SR.BID representation.

Another opportunity to enrich our understanding of APE is to expand the data set over which the SR.BID representation has been applied to include non-textual data, such as diagrams, verbal protocols, physical representations, and computer simulations. Already, SBF extensions exist for diagrammatic reasoning and computer simulations; similarly one could expand SR.BID in those directions, perhaps grounding it in those SBF extensions.

Another area for improved analysis of the APE phenomenon lies in not only the type of data but also the frequency with which it is collected. Observations of problem transformations “in real time,” such as might be found in shorter, design experiments using transcript and video analysis (e.g. the Delft design protocols) could (and almost certainly would) reveal transformation patterns that take place over much shorter time windows, which cannot be detected or inferred from data collected in three-week intervals.

While the studies here are limited to BID, there is no reason in principle why either SR.BID or the APE process should be limited to BID. Instrumenting other design environments, such as mechanical engineering classrooms, using tools such as problem definition assignments, one could perform parallel studies to identify changes in both quantitative and qualitative aspects of the APE process in other environments. One

imagines a greater influence of existing man-made solutions on problem formulation in a traditional mechanical engineering, than is seen in BID.

8.3.2 Problem definition through design lifecycle

The research in this dissertation is limited in scope to the concept development phase of design; the resulting theories and representations are almost certainly biased in this regard. For example, I site generally low occurrence of constraint concepts relative to function and operational environment concepts, in problem formulation. This would almost certainly change as the design progressed from conceptual to embodied design and physical implementation and testing, the hypothesis being that specifications and performance criteria will begin to dominate later stage thinking. Such a result would be interesting, when compared to results that show design thinking progresses toward function and behavior in later stages.

8.3.3 Expert versus novice studies

In the generalizations section I provide some thoughts on what makes for an “expert” in biologically inspired design. A lack of clear definition of the category notwithstanding, designers with more experience than the subjects examined in this dissertation may, and likely do, frame problems much differently than novice designers. With SR.BID, we have a systematic method for describing and classifying such differences for design problems.

8.3.4 Extended studies of the four-box and T-chart methods

As design tools, the four-box and T-chart methods are proven to be both usable and valuable in the context of the BID classroom. In principle, the four-box method is immediately extendable to any mechanical engineering context, which provides one

degree of generalization. While my studies demonstrate the perception of value from students, more rigorous studies of the effect of the four-box method on problem formulation, as well as down-stream design effects are necessary. At least one study is required to identifying differences in problem formulation under controlled conditions is necessary. Difference metrics should include conceptual content of the problem formulation, (e.g. SR.BID concepts), conceptual connectivity of the problem formulation, (e.g. SR.BID concept relationships) and (c) time spent on problem formulation. As low-hanging fruit, an exploratory comparison of content and connectivity between problem formulation assignments from BID classes for years pre- (2009, 2010) and post- (2011, 2012) introduction of SR.BID may provide some insight. The amount in variability between BID@GATech classes, may preclude strong conclusions.

Subsequent studies are suggested for understanding down-stream effects on common errors (e.g. analogical mismatch, vague problem definition, fixation), creativity, and feasibility. One of the challenges of such studies is to provide a sufficiently controlled environment to make comparison meaningful, with sufficient time allowance (e.g. a semester-long design project) to understand the longer-term design implications of structuring design problem thinking in this way.

8.3.5 T-chart as a tool for understanding analogies

The T-chart tool is examined in this dissertation, as an extension of SR.BID and the four-box method. The T-chart may also be considered as a window into the process of analogy making in BID. I have analyzed two design trajectories that use T-charts, the details of which are provided in Appendix L. In this analysis I found that in addition to providing students with a tool for analogical evaluation, T-charts provide insight into the

process of analogical problem evolution, and into compound analogical design. In particular in my analysis of the Green Light design case study, I show how analogies that appear grossly mismatched when viewed objectively, are in fact well matched, but for smaller sub-problems within the subjective context of the design team. This finding suggests how future tools and support systems must consider not only the problem as a whole, but design sub-problems as well when searching for and evaluating analogies. This finding also suggests that the T-chart could be enhanced to consider the more complex process of compound analogy.

8.3.6 Extension of the SR.BID Web Application

The SR.BID Web Application provides a platform for design, supporting design collaboration, through document and image sharing, project logs, and four-box problem definition. The platform also provides an online database of biological systems extendable by designers, and searchable using the SR.BID-based representations. The system is currently be integrated into BIDE, an larger collaborative environment that integrates the tool with DANE, Biologue and other tools developed at the Design and Intelligence Lab at Georgia Tech. In this dissertation I demonstrate that the SR.BID Web Application may be used by students to define problems and share information on biological systems. I also show that students are able to upload structured representations of problems and biological systems, useful for search, retrieval and analysis. The SR.BID Web Application provides the opportunity to study how SR.BID may be leveraged across both time (e.g. year to year) and space (e.g. different universities), to grow a sharable database of both biological solutions and related problems. Such a database would provide immediate value to the BID community, but would most certainly raise a number

of new research challenges such as curating the database, content ownership, etc.

The SR.BID Web Application also provides the opportunity to study how one may gather useful structured knowledge across a large number of independent users, at low cost, while delivering immediate value to the user. This is an essential question faced by many knowledge-based AI researchers.

8.4 Closing thoughts

Creativity and innovation are highly prized qualities in designs and the design teams that can generate them. The innovation leader, Apple Inc. provides an extreme example of the value of innovation. Prior to the release of its first iPod, Apple stock price was \$7.75 per share. In 11 years, fueled by the innovative iPod and subsequently breakthrough iPhone and iPad devices, Apple stock price was worth nearly 100 times that, over \$700 per share, growth nearly 80 times greater than the NASDAQ index over the same time period. Stock price is but one measure of value. The value of low-cost water delivery and purification innovations for third world countries can be measured in hours of menial labor saved and in the reduction of water-borne illness. The value of innovations in portable cold storage that enable the delivery of life saving vaccines to remote areas can be measured in lives saved. Whether portable music, or water purification, or the delivery of vaccines; at the heart of each innovation is the design problem to be solved. Interestingly, the final design problem solved is rarely, if ever, the same as the initial design problem posed. But what *is* a design problem?

Some argue that a design problem exists in the world as part of an objective reality, awaiting a designer to discover, understand and solve through some satisfactory design.

Others argue that a design problem exists in a completely conceptual, subjective realm, as unique as the designer conceptualizing it. Regardless of the epistemological perspective, one method of describing a design problem in terms of the *functions* and *behaviors* that design problems consistently exhibit. Functionally, a design problem plays at least three roles in design. First the design problem constrains and focuses, to greater or lesser degrees, the domain of the solution to be developed. Second the design problem serves as a means of evaluation for a solution, relative to some perceived need. Third, the design problem serves as a scheme to organize and index the memory of existing solutions that satisfy one or more aspects of the design problem.

Within the context of the first function, that of constraint and focus, the degree to which a design problem may constrain the solution is variable; it may, and usually does, go through periods of contraction (higher constraint) and expansion (lower constraint). Such fluctuations need not be constrained to a single domain, potentially providing access to highly disparate solution domains. Secondly, despite these “local” expansions and contractions, the overall behavior of the design problem in successful designs is to contract progressively toward a final solution.

Within the context of the second function, evaluation of a solution, the design problem provides representations over which evaluative functions may operate. These representations must enable evaluation such that the evaluation outcome correlates¹⁸ with the need being addressed. A design problem is a necessary condition for the operation of those evaluative processes (which may involve simulation, calculation, experimentation, analogy, etc.). The design problem also exhibits an extraordinary, well documented

¹⁸A simplification; the evaluation needs only be perceived to correlate with the need, such that a positive evaluation yields the perception that a solution will impact the need positively. Discussion of the evaluation machinery and its ability to accurately correlate with needs is out of scope.

behavior in the context of evaluation – it is subject to self-reflection and self-adaption. That is, the (cognitive) processes governing it are aware of some (or all) of its components and states, and capable of transforming it in response to (at least) evaluative feedback.

As a third function, the design problem provides a schema which may be used to access existing solutions. The design problem schema is so tightly integrated with solutions that designers often have difficulty cleanly distinguishing between the two. As a result, descriptions of design problems in terms of solutions are ubiquitous.

In the context of innovative design then, a design problem (a) constrains the solution domain, (b) aids in the evaluation of the solution, and (c) organizes and provides an index to the solutions and solution concepts available to the designer.

In this dissertation I demonstrate within the context of a class on biologically inspired design, and through a wide variety of studies the rich behavior of design problems over time. I firmly believe that innovation and creativity are intrinsically tied to human capacity to conceptualize of design problems differently; I believe that problem formulation, is a fundamentally distinguishing characteristic of human intelligence; I also believe that one of the core values of biologically inspired design comes from the reconceptualization of design problems. It is my hope in this dissertation to elevate design problems to a first class object of study to further innovation and creativity; to provide a foundation for the development of computational programs that can assess, support and even generate new design problems; and to enhance the practice of biologically inspired design by provide a new route to value.

APPENDIX A: Projected Growth of Biologically Inspired U.S. Patents

In 2006 Bonser wrote a paper predicting a Sigmoidal Curve distribution for US patents including the terms “biomimetic”, “bionic”, or “biologically inspired”. He projected that as of 2005, we were little more than half way through the current innovation cycle, based on his model. Bonser’s prediction used data through 2005.

First, I recreated his study. I searched the US Patent and Trademark Office online database (www.uspto.gov), using the same search terms. Table A-1 shows the number of individual patent issued, by year, through 2012 containing the keywords mentioned above anywhere in the patent filing. I used an MS Excel spreadsheet to solve for the logistic model using data through 2005, reproducing the equation shown in the paper. Next I added data from 2006-2012, and re-ran the analysis with the new data.

Figure A-1 shows the actual (blue line) number of patents issued 1985-2012 versus Bonser’s projected values (red line). The actual curve versus his predicted values is quite different. Using the new equation to re-projected the values I show in A-2 the expected growth through 2040, with approximately 10 times more patents estimated than Bonser’s original projections. Table A-2 compares the original values against the re-projection values every 5 years from 2020 through 2040, after which time both curves remain relatively flat.

The new data show a greater acceleration in the growth of biomimetics than originally predicted.

Table A-1. Number of US patents containing the words “biomimetic,” “bionic,” or “biologically inspired”, issued 1985-2013

Year	Patents	Year	Patents	Year	Patents
1985	5	1995	116	2005	876
1986	9	1996	151	2006	1046
1987	16	1997	182	2007	1200
1988	22	1998	229	2008	1361
1989	25	1999	283	2009	1578
1990	33	2000	354	2010	1927
1991	43	2001	451	2011	2293
1992	53	2002	538	2012	2765
1993	66	2003	669		
1994	91	2004	776		

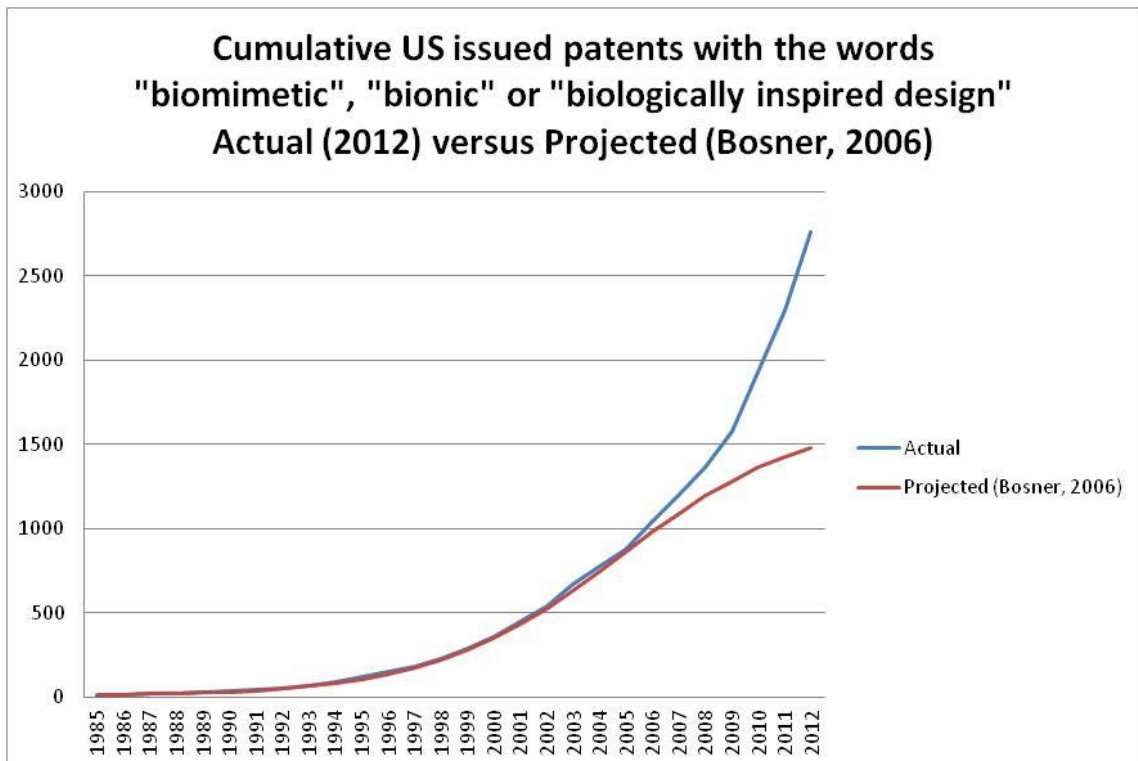


Figure A-1: Cumulative total of US issued patents 1985-2012.

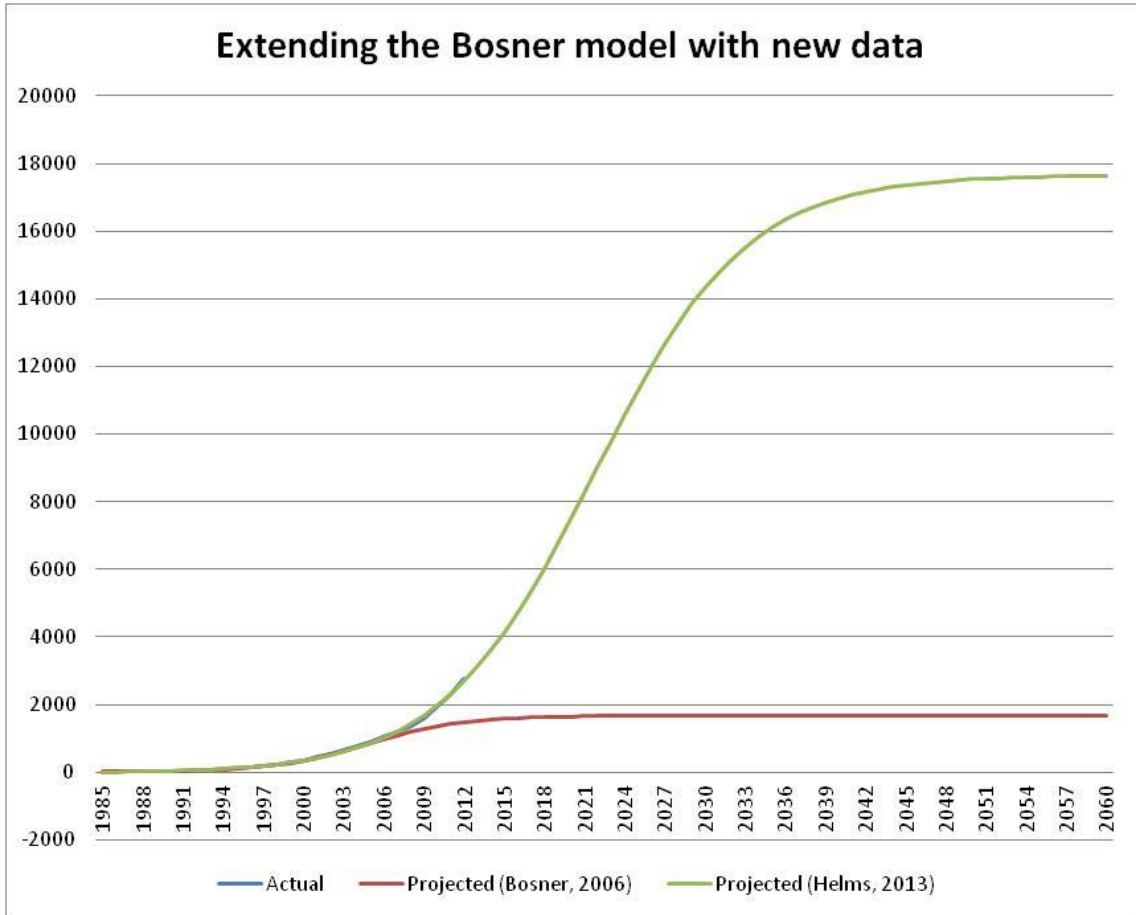


Figure A-2. Re-projection of BID patent growth, using US patent data through 2012.

Table A-2. Re-projection values vs. original projections.

Year	Bosner Projections	Helms Projections
2020	1649	7492
2025	1665	11304
2030	1670	14320
2035	1671	16099
2040	1671	16973

APPENDIX B: Examples of Compound Analogy

Several examples of compound analogy were observed in the 2006 class on biologically inspired design. Table B-1 provides a summary of the compound analogies, including the high-level design goal, and biological systems that served as analogues for part of each design. Additional details about each project are available in Vattam, Helms & Goel (2007).

Table B-1. Compound analogy projects.

Design Name	Description	Biological Analogues
BioFilter	Portable, stand-alone air-filtration system.	Diatoms, Spider Silk
BriteView	Monitor screens that remain visible in strong sunlight.	Hummingbirds, Butterflies
Eye in the Sea	Underwater, stealth micro-robot	Squid, Copepod
InvisiBoard	Surfboard that does not produce a silhouette when seen from below to prevent shark attacks	Pony Fish, Brittle Star
iFabric	A thermally responsive adaptive fabric for clothing, providing thermo regulation for the wearer.	Bee Hive, Artic Wolves
RoboHawk	Aerial bomb detection device	Seagull, Dog

I apply the conceptual framework of *compound analogical design* developed in the previous section to analyze two of the compound designs observed in the 2006 *in situ* study.

B.1. Eye in the Sea

The goal of this project was to design a small underwater robot with locomotion modality that would ensure stealth. The initial research for the underwater robot focused on the copepod (a small crustacean, 1-2 mm in length) as a source for understanding stealthy locomotion. In exploring this concept, designers became aware that the copepod used two distinct rhythms of appendage movement for achieving motion underwater. A

slow and stealthy rhythm was used when foraging for food and a quick but non-stealthy rhythm was used when escaping from predators. This understanding led the designers to decompose their original problem into two separate functions, one for slow movement, and one for rapid movement, both of which required stealth. *This new problem decomposition was based solely on the understanding gleaned from the copepod analogy.* The knowledge of the slow, stealthy mechanism used by the copepod, known as a “metachronal beating pattern” was also transferred from the copepod source.

Next, the designers had to address the second sub-function: fast, stealthy motion. They identified squid locomotion as an inspiration for achieving this function. The squid mechanism, jet propulsion, is both much *faster* and *stealthy*, matching its wake with external disturbances that naturally occur in the surrounding water. Notice the stealth achieved here (wake matching) is different from the stealth achieved by the copepod (wake minimizing).

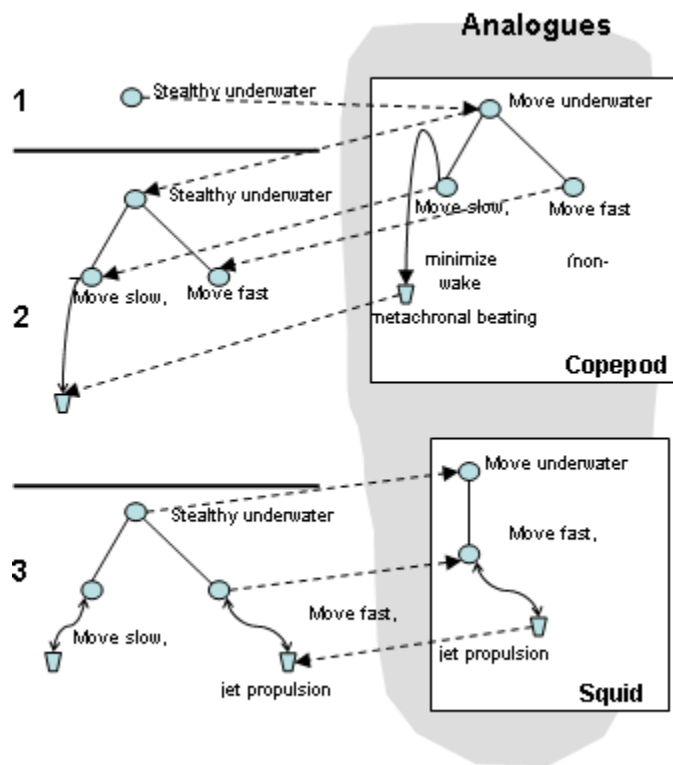


Figure B-1. Generation of compound analogy for Eye in the Sea

Fig. B-1 develops a model of the generation of this solution using the framework of the compound analogical design. Step 1 depicts the nature of the problem space early in

the design. The main function is to move underwater stealthily, and the copepod is identified as a solution analogue. In Step 2, based on knowledge from the copepod analogy, the function of moving underwater is decomposed into sub-functions. The solution to the function of moving slowly by minimizing wake is adapted from the copepod to generate a partial solution. But the function of moving fast, yet stealthily remains unresolved in Step 2. In step 3, the squid analogue is retrieved to address this function. Its solution of using jet propulsion for movement is transferred to the current problem to generate the remaining solution. These two partial solutions are aggregated to achieve the trial design.

B.2. InvisiBoard

The goal of this project was to conceptualize a surfboard that minimized its silhouette to prevent “hit-an-run” shark attacks. Designers chose the pony fish, which produces counter-illumination by producing light that is directly proportional to the amount of ambient light, as their source of inspiration. Using this analogy, the function of silhouette

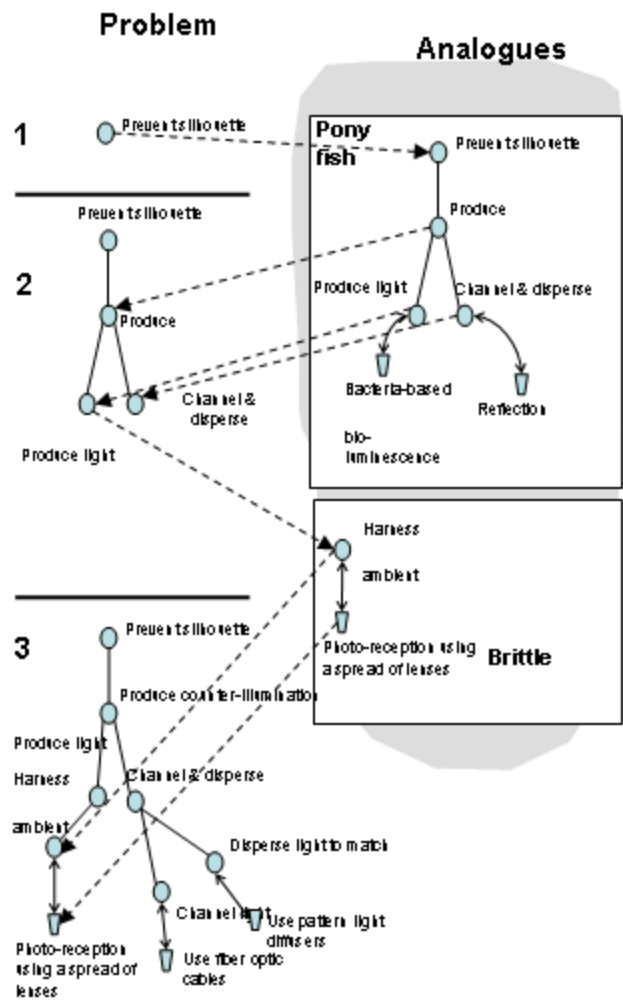


Figure B-2. Generation of compound analogy for InvisiBoard

camouflage required the sub-function of producing a glow on the ventral side of the surfboard to match the ambient light. Based on a more detailed understanding of the mechanism employed by the pony fish, function of producing ventral glow was decomposed in the sub-functions: produce light, channel and disperse light.

In order to produce light, a light source and a power source onboard the surfboard was considered an inferior solution. The search for alternate means of producing light led them to an organism called a brittle star (a kind of a star fish). The dorsal side of the brittle star is covered with thousands of microscopic lenses. This suggested a design in which the top of the surfboard would be covered with (suitably distributed) lenses to collect the sunlight incident upon the surfboard. Dispersion of the light would occur via optic fibers. Additional design steps include adding a layer of “pattern light diffusers” on the bottom of the surfboard, which disrupts the pattern of light from the optical fibers to mimic the wavy pattern of the ocean surface. Fig. B-2 demonstrates the generation of this solution using the framework of the compound analogical design.

APPENDIX C: Sample Problem Decompositions

The following is a sample of problem decompositions used for training purposes in the 2007 study of student problem decompositions.

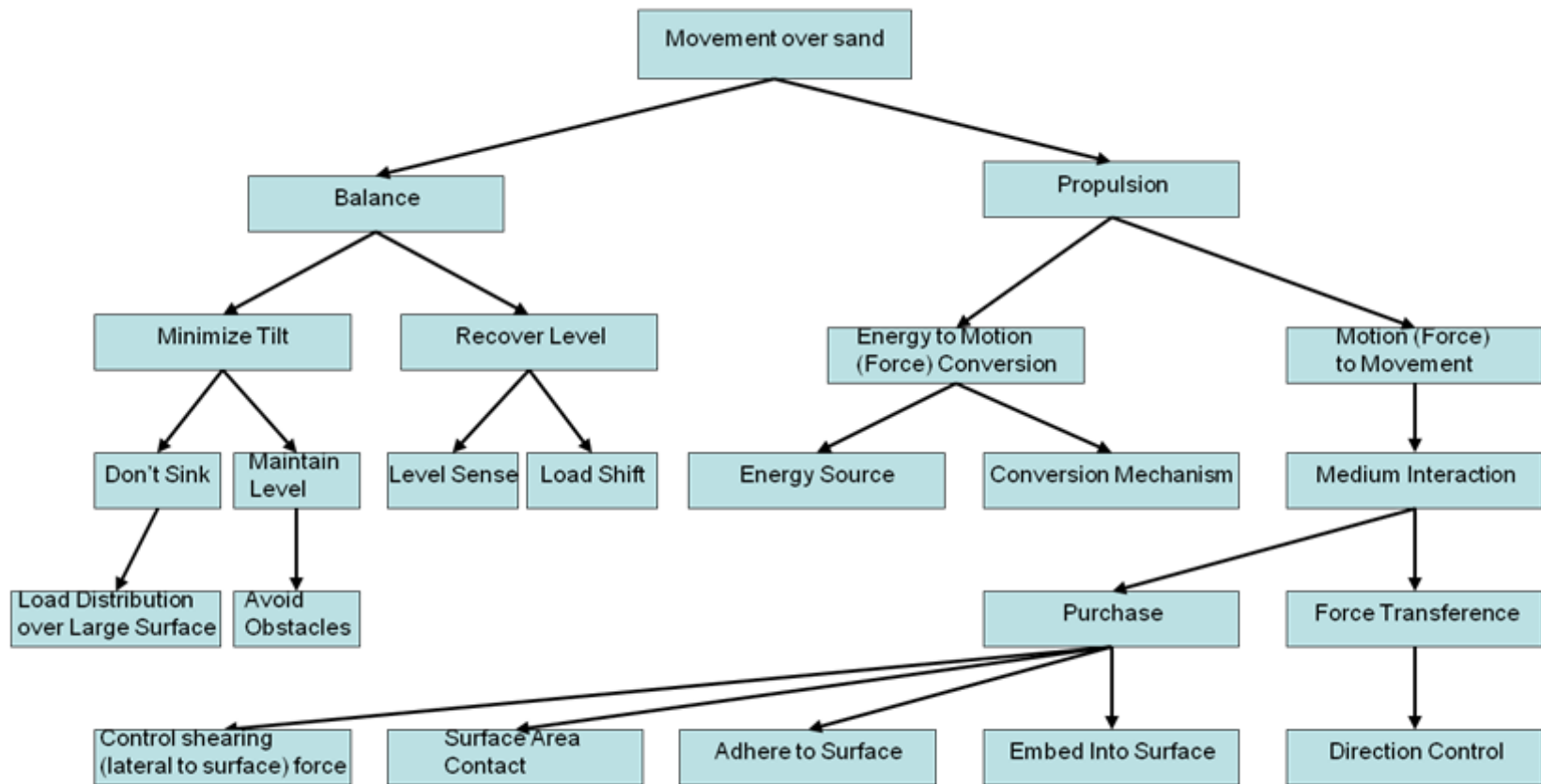


Figure C-1. Functional decomposition for movement over sand, created by instructors.

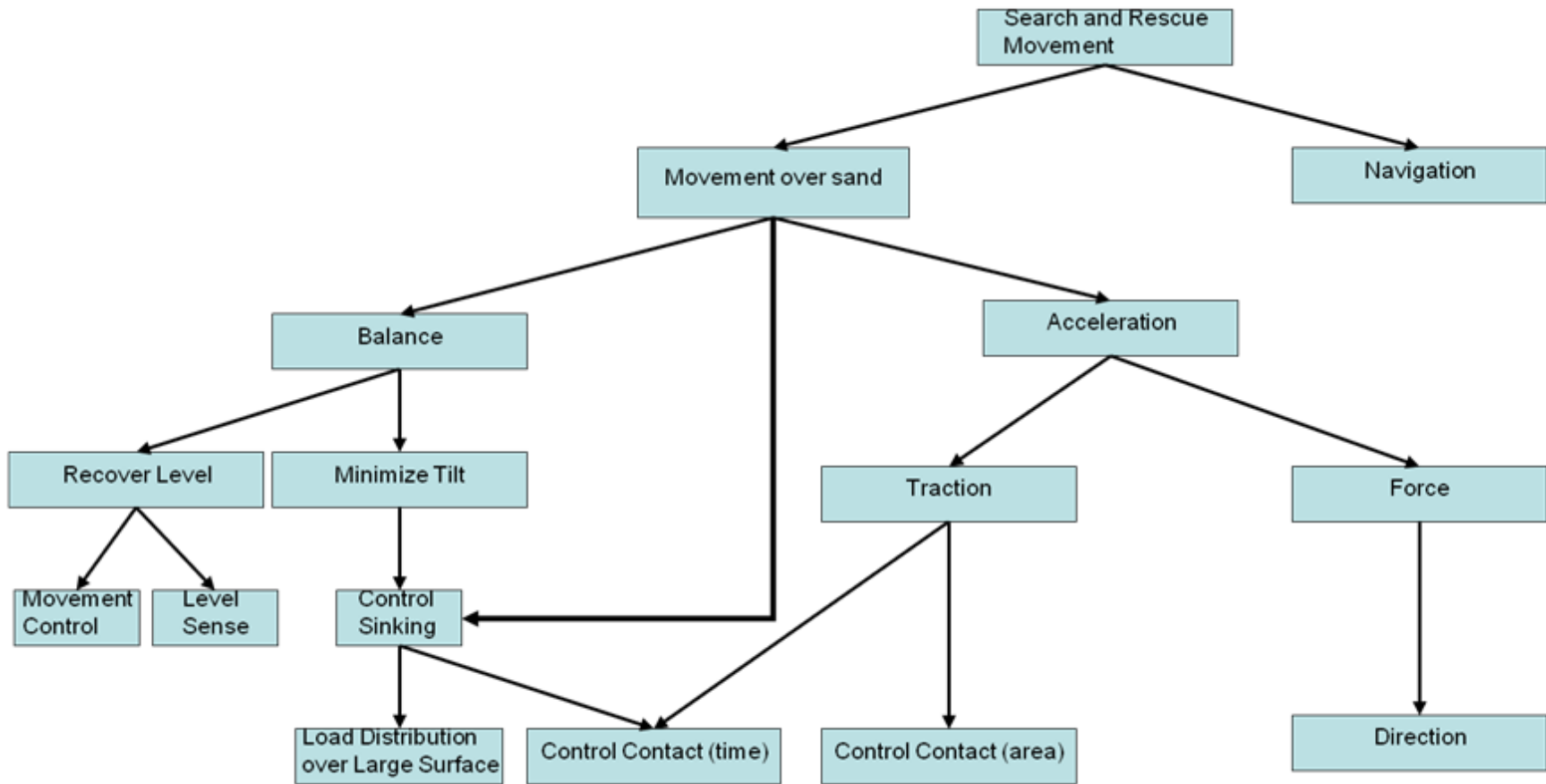


Figure C-2. Functional decomposition for movement of sand, created by students

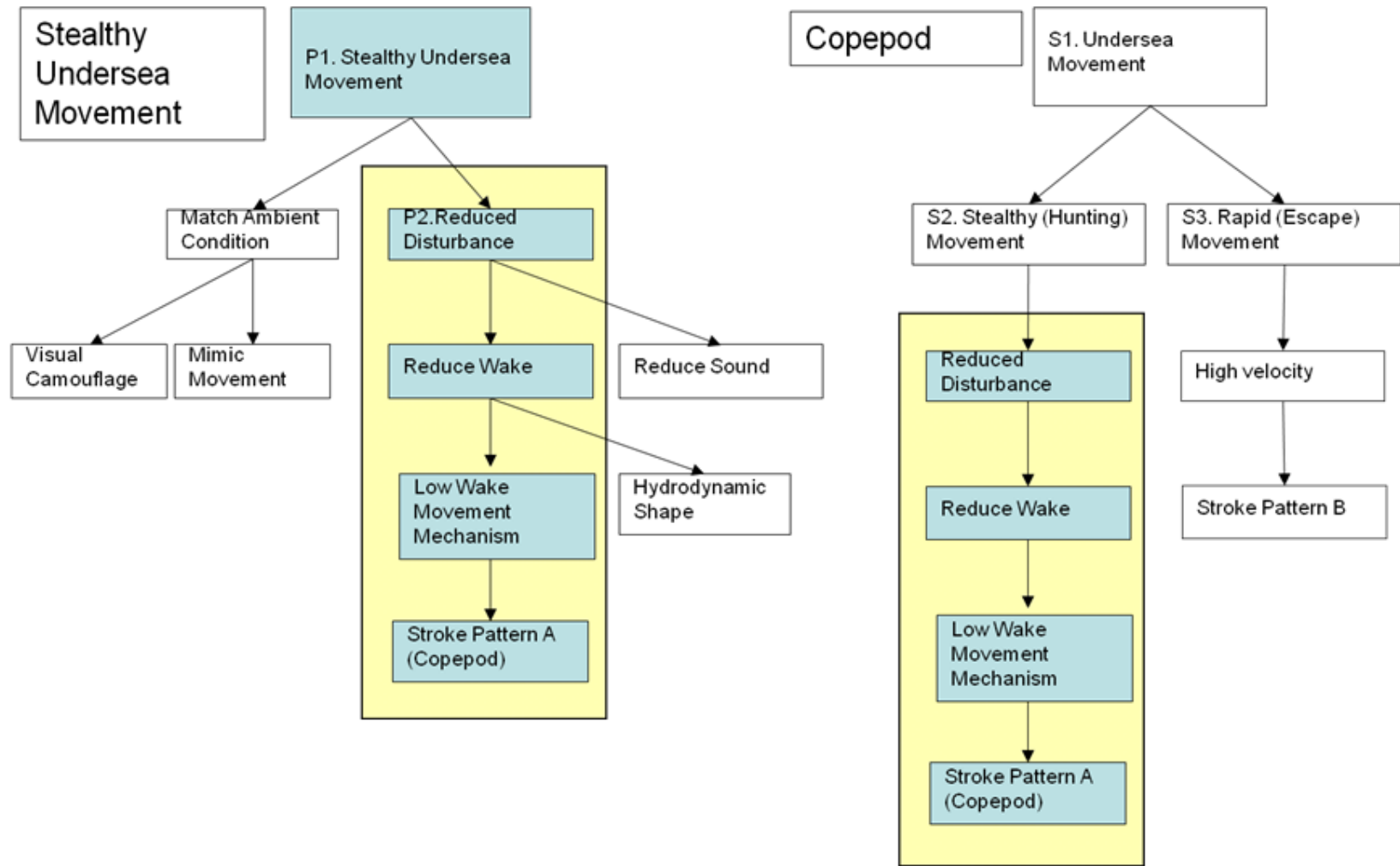


Figure C-3. Functional decomposition for stealthy undersea movement and copepod.

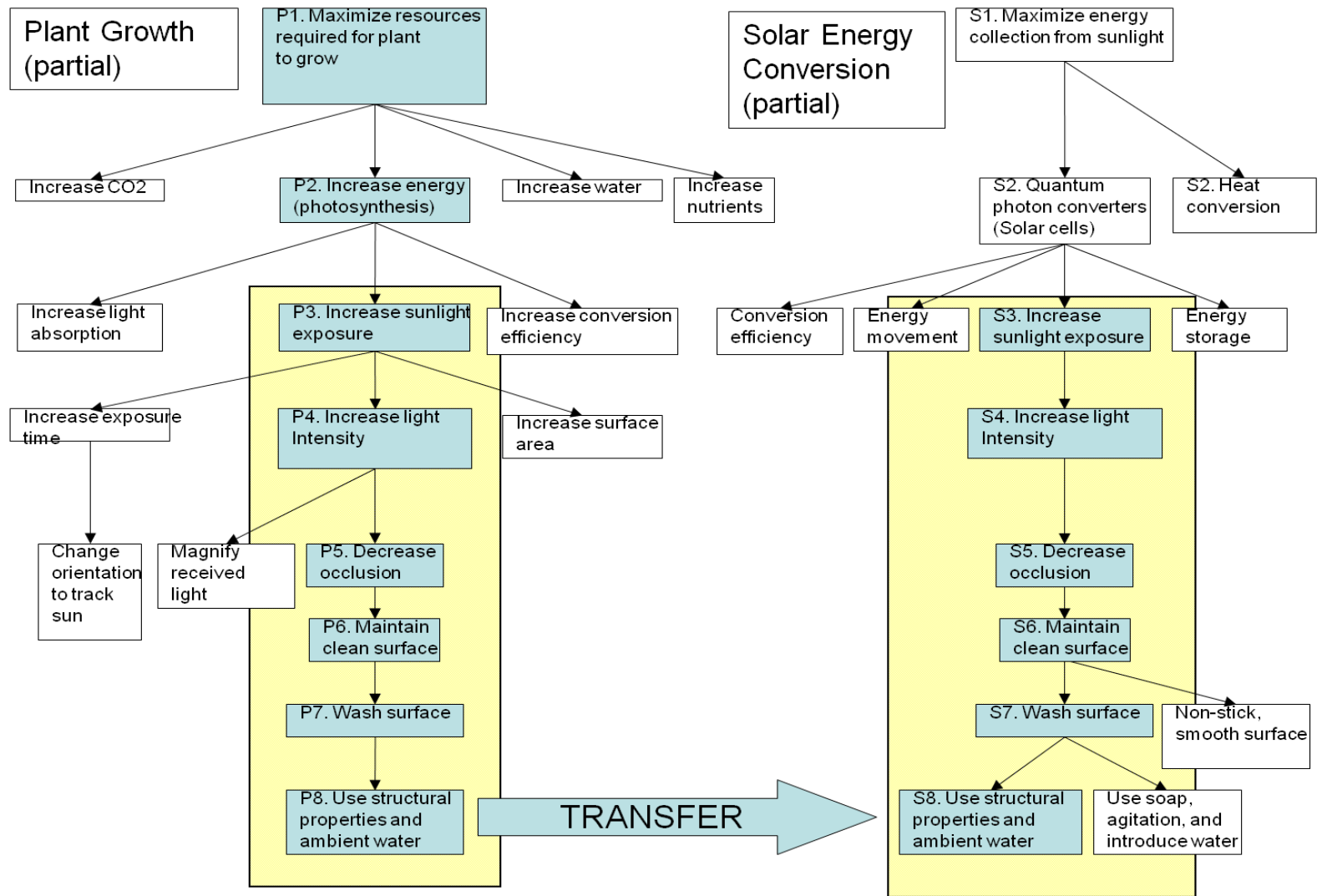


Figure C-4. Functional decomposition of plant growth and solar energy conversion.

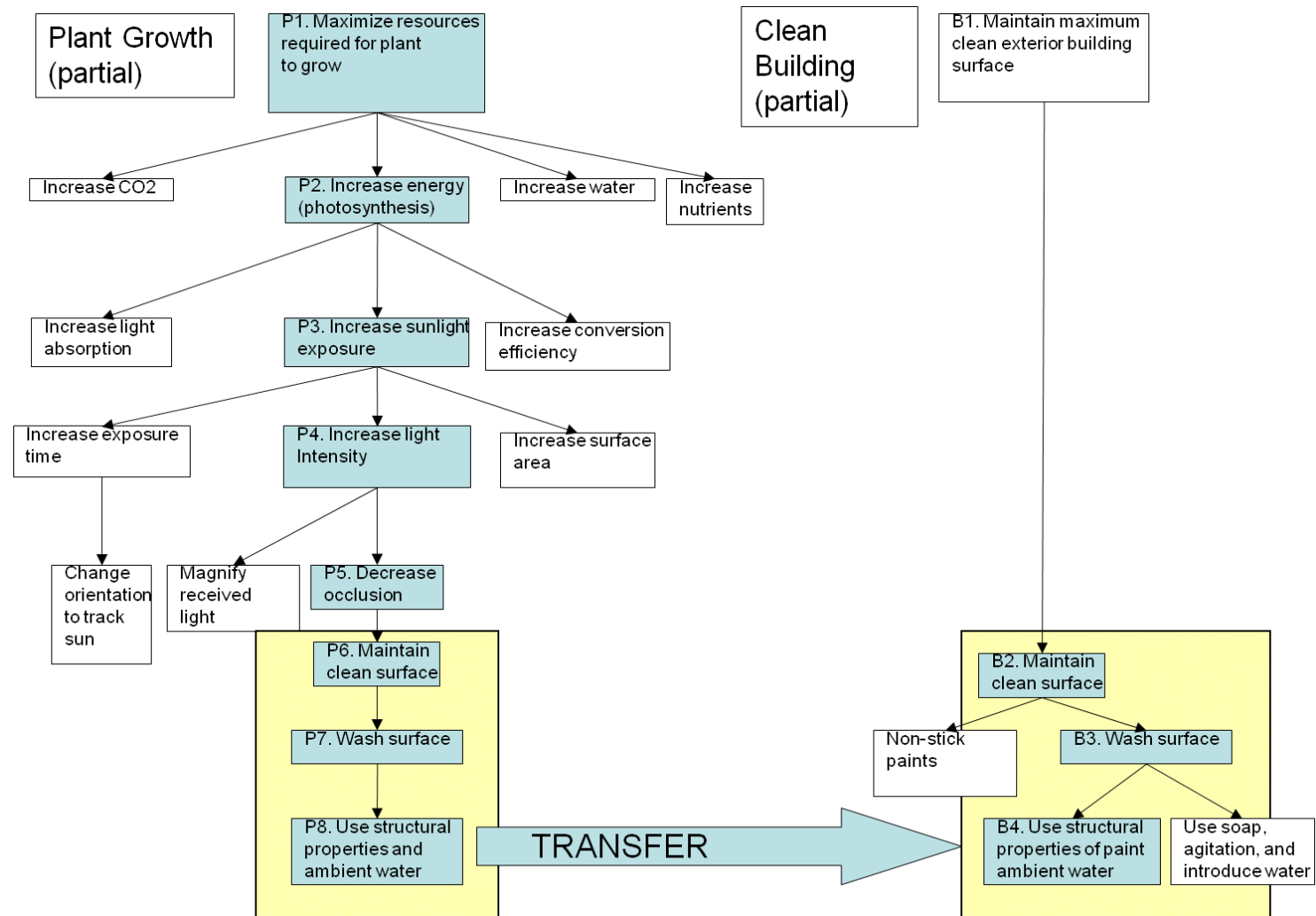


Figure C-5. Functional decomposition of plant growth and clean building.

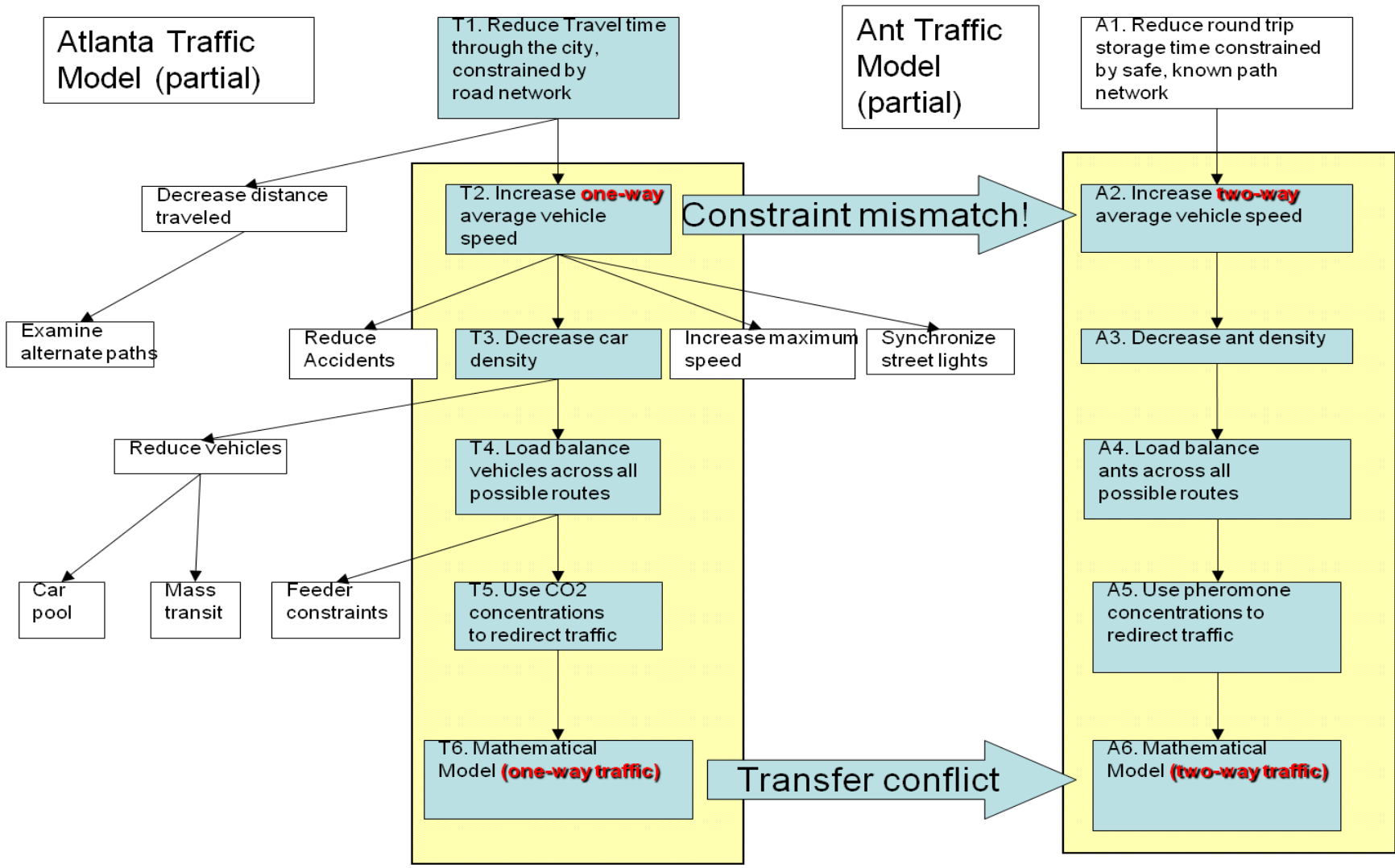


Figure C-6. Functional decomposition of Atlanta traffic model and ant traffic mode, showing constraint mismatch.

APPENDIX D: Biological System Information, Study E2

This appendix provides the content of the biological analogues provided to students during study E2. This is the actual content, although some margin and font formatting changes may have occurred to align it with the format of this dissertation.

1. *Mussels*

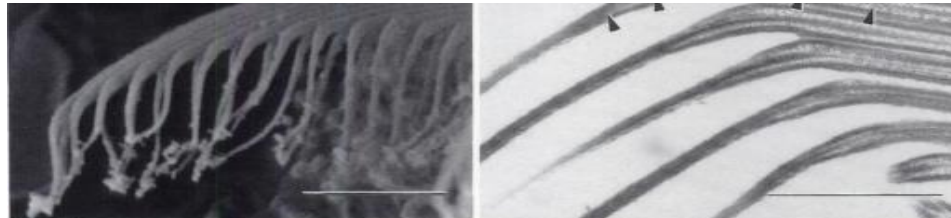


Figure D-1. Electron micrographs of the shoulder region of cirrus

Microscopic techniques used to examine the filtration mechanics employed by the zebra mussel, *Dreissena polymorpha*, reveal that the organ primarily responsible for particulate capture is the gill. The gill is of classic eulamelibranch form with two epithelial lamellae surrounding a central water channel. Particles are captured from water that is driven across the gill filaments with the aid of the lateral cilia on each side of the filaments. Latero-frontal cells and their complex motile organelles, either cilia or cirri composed of many fused cilia, are located between the frontal surface of the filament and the lateral ciliated cells (shown in **Figure D-1**). It is the location, orientation and dynamics of these latero-frontal cirri that have been attributed to an efficient participation in the interception of particles. The cirral cilia move together from a flexed position to an extended position with the cirrus projected in the plane of the latero-frontal cell and across the interfilament space. In the final position the free ciliary ends of the opposing and neighboring cirri form a sort of sieve or net with a spacing of about 0.2-0.7 μm . [Silverman et al, 1996]

The function of cirri in particulate interception has been attributed not only to mechanical filtration, using the “net”, but also to complex current formation and overall gill hydrodynamics, thereby capturing particles without actually

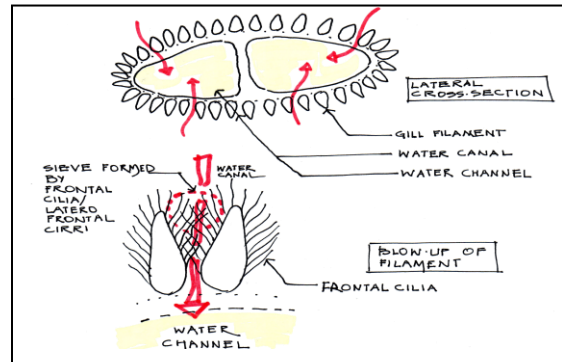


Figure D-2. Particulate capture

physically trapping them (**Figure D-2** shows the process). The functions are not mutually exclusive rather it is the combination of these different mechanisms that allows the D. polymorpha to retain particles smaller than 1 μm with over 90% efficiency. Particles filtered include single celled algae, diatoms, and some bacteria, which form the primary food source for the zebra mussel.

Although Mussells provide an efficient model for filtration, but the ciliary and complex hydrodynamic motions are very difficult to replicate with the available contemporary technologies.

2. Whales

Some whales, such as the bowhead whale shown in **Figure D-3**, use baleen to filter feed. They open their mouths to take water and swarms of krill into their mouths, then close their mouths and push the water back out, trapping the krill in their baleen. The “design” of the mouth of the baleen whale is such that



Figure D-3. Bowhead Whale

an accelerated flow is created which prevents the krill from escaping. The water with the krill enters the opening at the front of the mouth and slows as it enters the large cavity of the mouth, i.e. it undergoes a pressure differential. The pressure difference causes some of the krill to drop down out of the flow. The rest get caught in the baleen curtains at the sides of the mouth as the water is pushed out the back corners of the mouth. A summary of this process can be found in **Figure D-4**. The flow was analyzed by the Bernoulli, Hagan-Poiseuille, and Navier-Stokes equations, and all resulted in a similar order of magnitude of pressure drop. Pressure differentials were found to increase as speed of the whale increased (Werth, 2004).

Krill range from 8-60 mm in size. We would like to capture particles .3 microns in size, or at the very least 10 microns. The scaled model of the baleen whale’s mouth followed the mathematical model reviewed in the paper. Therefore, it

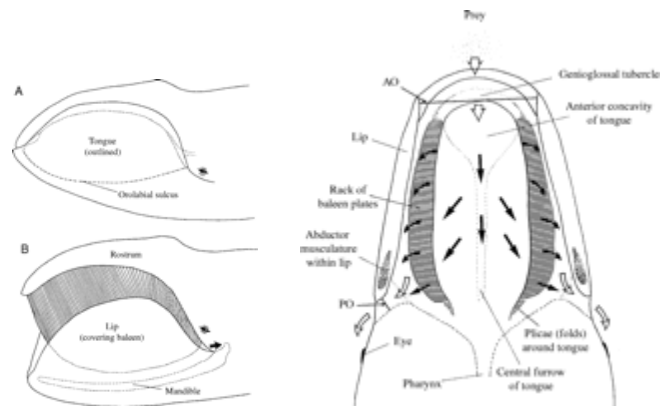


Figure D-4. Filtration process of a bowhead

is possible that we could model the baleen whale’s mouth to such a scale that it would

capture particles of the desired size. Using the combination of accelerated flow, resulting pressure differential, and a baleen-like screen in a compact filter could result in effective filtration of air in a room.

Thus a baleen whale-based system would be relatively simple to create, and the model made in the paper should be replicable; however, it is unclear whether or not the model would be scalable to the size necessary (i.e. it would only filter some, not all, of the particle sizes we want to target). Also, there would need to be another mechanism to create the air flow through this filter, since in nature the whale is moving to create the flow.

3. *Diatoms*

Diatoms do not filter fluids in their environment, but their structure is such that they would make excellent filters. Diatoms have regularly spaced, uniformly sized pores in their cell walls, or frustules, which they make out of silica. Since they are microscopic (up to 500 microns in diameter/length) and have pore diameters ranging from 20 to 200 nanometers, they are just the right size for filtering out the smallest particles we want to remove from the air. They also replicate very quickly and come in a variety of shapes; so we could choose the size, shape, and pore size of the diatoms that would work best as a filter, grow them, remove the organic material, and use the frustules as the filter media.

Parkinson & Gordon actually suggest using diatomaceous earth (pieces of frustules randomly oriented & packed), but they are simply wanting to separate large particles from small ones by making them flow at different rates; we want to ensure that all particles are captured. Parkinson & Gordon also note that using diatoms in filtration is “cheap,” which is a trait we desire.

4. Salps / Larvaceans

Cyclosalpas and Larvaceans (Figure D-5) are both jellyfish-like organisms that utilize mucus nets to filter out food from their environment. Both are very adept filter feeders. The cyclosalpa in particular uses a mucus net that extends into the pharyngeal cavity, along the gill bar and into the stomach. This

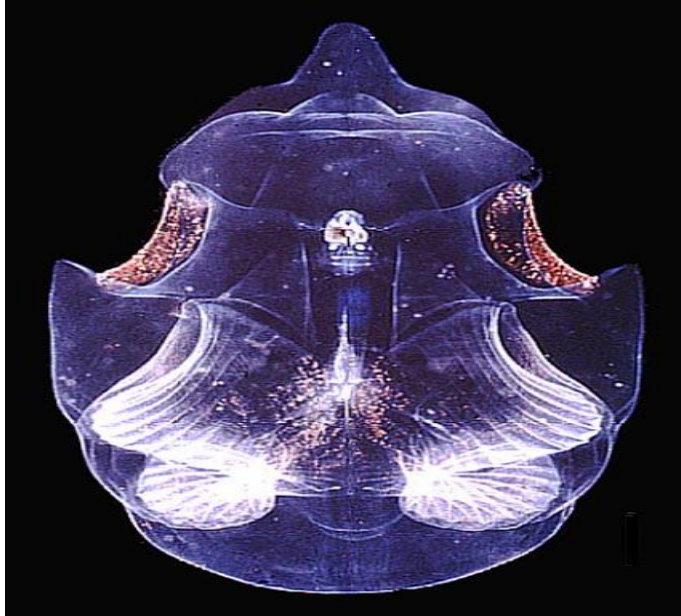


Figure D-5 Cyclosalpas/Larvacean

This mucus net is continuously produced by endostyles. Gill cilia pump water through the mucus net as the salps move through the water using rhythmic contractions of circular muscles on their body walls. All of the water that passes through a salp is filtered through the net. The continuous production of the mucus net allows the salp to feed continuously while renewing the net. With their efficient filtering mechanism, dense aggregations of this species can quickly deplete phytoplankton within a wide area^{5,7}.

The food particles that these organisms are after are typically phytoplankton that range in size from 0.7 to 100 microns. Salps and Larvaceans are nearly 100 percent efficient at capturing particles above the 3 micron size^{5,6,7}.

Converting this mucus net design into a functional system would be both simple and difficult. On one hand, the design concept is simple enough: a hole lined with a mucus-net. This filter would be able to filter out a wide range of particulate sizes, from 0.7 to

100 microns. And it would be highly efficient, nearly 100 percent efficient for particles greater than 3 microns. However, this system would be difficult to maintain since the filtering efficiency of the mucus depends on its moisture content. One option would be to provide a constant flow of mucus, but that would require the added complexity of a pumping system, as well as a mucus filtration system. Even if the mucus were made to be gel-like instead of flowing, there would still have to be some separate system that keeps the gel medium humid.

5. Hemoglobin

One common type of filtration done in nature is often overlooked, because it is not the filtration of air or water, but rather gasses from the air. Hemoglobin, found in the blood of most mammals and also other animals, has the capability of binding oxygen very strongly out of a variety of gasses present in the air.

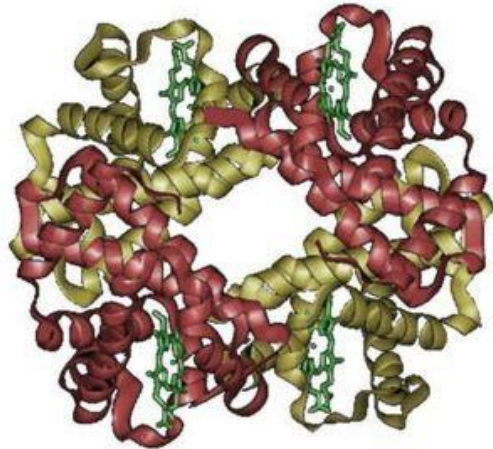
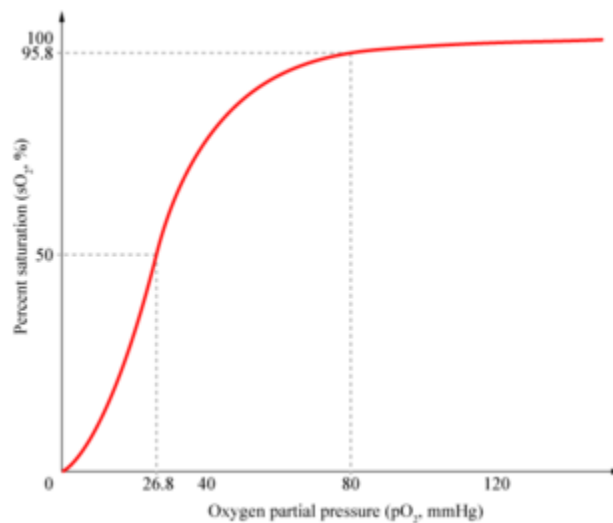


Figure D-6. Hemoglobin

The hemoglobin molecule, as seen in **Figure D-6**, is comprised of 4 sub-groups, each with a globular protein containing an embedded heme group. This heme group contains an iron atom which gives it the ability to bind oxygen. Thus, each hemoglobin molecule in a human can bind 4 oxygen molecules.

Furthermore, the binding of hemoglobin is cooperative, meaning that as each oxygen molecule binds, the affinity of the hemoglobin for oxygen increases exponentially. A typical binding curve is shown in **Figure D-7**. In mammals, hemoglobin is used to transport oxygen from the lungs to the various tissues in the body.



FigureD-7. Cooperative binding

There are also several variations present in nature which operate on the same principal, such as myoglobin (single sub-unit meant for oxygen storage) and hemocyanin (uses a

copper group instead of iron to bind). Interestingly enough, one variation of this molecule, leghemoglobin, is found in leguminous plants, such as alfalfa or soybeans, and is capable of binding oxygen to *protect* the nitrogen fixing bacteria in the roots. One of the main weaknesses of hemoglobin is its affinity for carbon monoxide, which competes for the same binding site as oxygen, but has a 200 times greater affinity. This makes carbon monoxide very dangerous for humans in even low concentrations because it has the ability to starve the body of oxygen.

APPENDIX E: SR.BID Sub-category Descriptions

The following tables provide a description of the sub-types used for coding problem statements in the Week 3 2010 and Week 8 2010 data sets.

Table E-1 SR.BID primary and secondary solution types

SOLUTION	Description
Primary Type	
Biological	The solution is a naturally occurring biological component, organism or system.
Man-Made	The designers refer to a system which someone already built or created, or for which they generated prototypes or specifications.
New Design Solution	The designers who are working on the problem are conjecturing a new design (or a design they think is new) to solve the problem.
Secondary Type	
Sub-Solution	A sub-solution consists of many parts that together perform a specific function within the context of a larger solution.
Sub-Type	A sub-type solution expresses a "kind-of" relationship with another solution.

Table E-2 SR.BID primary and secondary function types

FUNCTION	Description
Primary Type	
Accomplishment	The default function type, accomplishment functions change the state of the world in an intended way.
Preventative	Preventative functions keep a state OR another function from occurring.
Maintenance	Functions that maintain a state are considered maintenance for example "the thermostat regulates temperature" is a maintenance function.
Allow	Allow functions enable a state or another function to occur.

Table E-2 continued

Negation	Negative functions are stated as NOT performing another function, for instance "this application does not produce light."
Secondary Type	
Sub-function, AND	When there are multiple sub-function relationships for a given function, AND-type relationships that specify that the related sub-functions must all be accomplished in order to achieve the parent function.
Sub-function, OR	When there are multiple sub-function relationships for a given function, the OR-type relationship specifies that one of the functions must be accomplished to achieve the parent function.

Table E-3 SR.BID primary operating environment types

OPERATING ENVIRONMENT	Description
Primary Type	
Location	The places in which the system is intended to operate.
Condition - Qualitative	Qualitative conditions under which the system is intended to operate.
Condition - Quantitative	High/low end values, expected values, or ranges.
Time	The time during which the system must operate for example, "at night." Words like "when", "after", "while", "as" and "during" are often used to express a temporal environment.
User	The phrase describes an intended user or class of users for the system.
Entity	The phrase describes an entity, often biological but sometimes technological, that interacts with the system
System	The phrase describes another system within which the system is intended to work or connect.

Table E-4 SR.BID primary and secondary constraint and specification types

CONSTRAINTS & SPECIFICATIONS		Description
Primary Type		
Material		The material of which one or more components of the design will be composed.
Information		Information can be in the form of energetic signals, bits and bytes, or may be encoded in the physical structure of a thing.
Energy		Energy can be found throughout a system in many forms; the energy sub-type is used when a specified form of energy is discussed within the confines of the system.
Time		Includes timeframes not related to the operation of the design.
Component		Includes descriptions of specific parts of a solution or design, or groups of parts.
Property/Value		Concerns the properties of the system as a whole or their values.
Shape		Includes the shape of the components or of the design.
Spatial Orientation		These specify the spatial relationship or orientation between or among one or many components, systems, or sub-systems.
Structural Relationship		Any phrase specifying which components are related by means of connecting joints and contacts points.
Cost		Usually in monetary terms, but this could also be in terms of any resource of concern; absolute or relative.
Secondary Type		
Limiting		Limiting specifications/constraints are those which require a designer to use a smaller subset of design elements.
Enabling		Enabling specifications/constraints offer new possibilities for design elements without enforcing their use.
Existing		Existing specifications/constraints discuss the specific properties of an existing design.

Table E-5 SR.BID primary performance criteria types

PERFORMANCE CRITERIA	Description
Primary Type	
Specific	States the specific value or range of the performance criteria.
Relative	Uses comparative terms such as "quieter than solution X", without explicitly stating the performance of the compared to solution.
Actual	States the performance of an existing solution.

Table E-6 SR.BID primary deficiency and benefit types

DEFICIENCY/BENEFIT	Description
Primary Type	
Deficiency	Deficiencies can relate to any element of an existing solution or proposed design, highlighting an unfavorable aspect of that element.
Benefit	Benefits can relate to any element of an existing solution or proposed design, highlighting a favorable aspect of that element.

APPENDIX F: SR.BID Coding Rubric

All text and transcript documents were encoded into an SR.BID problem schema using the following rubric. Each coder used the rubric as a general guideline when coding text. The rubric provides enhanced definitions and examples for each category and sub-category, as well as the notation used for the coding. A “what to watch out for” provides advice for common patterns that may not be immediately obvious, or for which mistakes are commonly made.

Code Form	Question	SOLUTIONS Criteria	Examples
	Does the phrase refer to a SOLUTION?	Solutions are man-made systems, organisms or conjectured systems that perform a useful function. Solutions are almost always nouns, usually constructed devices or organisms. Sometimes solutions can be a combination of man-made system and organism (a plow pulled by an ox, for example). Sometimes the solution can be non-physical -- software, or an algorithm.	"eagle eye", b(1): eagle eye; "coffee mug", s(1): coffee mug; "fusion reactor", sd(1): fusion reactor,
SUB-TYPES (Biological, Existing, New Design)			
b(x)	Is the solution BIOLOGICAL?	Is the solution a naturally occurring biological component, organism or system? Biological systems may occur at varying scales from molecular (proteins) to tissues (cardiac muscle) or organs (heart) to systems (cardiovascular system) to ecosystems (rainforest).	"cell wall", b(1): cell wall; "ankle joint" b(2): ankle joint; "flock of seagulls"; b(3): flock of seagulls
s(x)	Is the solution EXISTING and MAN-MADE?	Are the designers referring to a system that someone built, created or generated prototypes or specifications for already?	"HVAC system", s(1): HVAC system; "shoe laces", s(2) shoe laces; "velcro fly", s(3): velcro fly.
sd(x)	Is the solution a new SOLUTION DESIGN?	Are the designers who are working on the problem conjecturing a new design (or a design they think is new) to solve the problem?	"underwater micro-robot", sd(1): underwater micro-robot; "hawk-nosed bomb detector", sd(2): hawk-nosed bomb detector; "invisible surfboard", sd(3) invisible surfboard.
SUB-TYPES (Sub-Solution, Sub-Type, Sub-Component)			
[b, s, sd](x-n, s)	Is the solution a SUB-SOLUTION of another solution? (sub-type s)	A sub-solution consists of many parts that together perform a specific function within the context of a larger solution. For example a transmission is a sub-solution of the power-train which is a sub-solution of an automobile. We code this example s(1):automobile, s(1-1, s): power train, s(1-1-1, s): transmission. The automobile also has an electrical system, which in this case we would code as s(1-2, s): electrical system.	"a tree has roots a trunk and leaves", b(1): tree, b(1-1, s):roots, b(1-2, s):trunk, b(1-3, s): leaves
[b, s, sd](x-n, t)	Is the solution a SUB-TYPE of another solution? (sub-type t)	A sub-type solution expresses a "kind-of" relationship with another solution. For example "manual transmission" and "automatic transmission" are two kinds of transmission. We code this example s(1): transmission, s(1-1, t): manual transmission, s(1-2, t): automatic transmission.	"we are building a single device, and we could make either a morpho-butterfly light reflector or a bull-frog noise maker" sd(1): signal device, sd(1-1, t):morpho-reflector, sd(1-2, t): bull-frog noise maker.

What to watch out for.

Natural occurring phenomena are not solutions for example "rain cloud" is not a solution to "deliver water". There may be underlying phenomena about rain clouds --"evaporation", "cohesion", "condensation", and "thermal gradients" -- which are useful for creating new solutions, but which themselves are not considered solutions.

People, either individuals or groups, are not solutions (per se, they may be components in a larger system, however, like a bucket brigade). Organizing them to take collective action may be a solution, parts of them may be considered solutions to problems (kidney is a solution to filtering blood), but the people themselves (scientists, teachers, etc.) are not solutions to engineering design problems (they may be solutions to other kinds of problems).

Some systems and organisms that are mentioned are not mentioned in the context of a "solution" but rather in the context of "things that interact with the solution being designed". These belong in the operational environment section as reactive entities.

This MAY or MAY NOT include bioutilization systems, which are man-made applications that utilize naturally occurring biological organisms. If the phrase is primarily about the organism or a single kind of organism in the context of a bioutilization system, then it is a BIOLOGICAL system. If the phrase is primarily about an human-constructed complex SYSTEM of many ORGANISMS used to perform some task then it will fall into the man-made category. "Fungi" is a biological system; "Mycoremediation" is a man-made system that uses fungi to eliminate toxin material.

Designs being worked on are considered EXISTING solutions, e.g. "Scientists are working on building ultra-lightweight flying robots" is s(1): ultra-lightweight flying robot

Often designers will refer to a proposed solution in vague terms for example "the solution must be small", which we code as sd(1): new solution; if the solution is based on an organism as in "the ant-based solution will help relieve traffic congestion", we can code it with a little more specificity, including the organism and something about the function sd(2):ant-based traffic solution.

A new solution may be an off-shoot or an improvement to an existing solution. It may be proposing a slight modification to a larger system that stays the same. Proposing a new tread pattern on a car tire, for example, is not a complete redesign of the car tire, but is a new solution with respect to the tread design. This can be a little confusing with respect to solution/sub-solution relationships -- a proposed new solution design can be a sub-solution to an existing solution (although this is rarely expressed in practice).

In bio-utilization, biological systems (b) are used as sub-components of a man-made system (s or sd). This complicates coding. Need to make the relationship with the super-system explicit as in s(1): mycoremediation; b(1, s)|s(1): fungi

FUNCTION			
Code Form	Question	Criteria	Examples
	Does the phrase refer to a FUNCTION?	Functions concern the actions that a system or design must perform. A phrase that is a function always contains an action verb (walk, filter, change, etc.), usually implies or states the subject performing that verb (lizard, kidney, enzyme), and sometimes states one or more objects to which that verb is applied (blood, lactose). The object of the verb can also be the subject. In the phrase "the lizard walks" the lizard (subject) performs the action (walks) to itself.	"the lizard walks", s(1) lizard, f(1) s(1): walks (self); "kidney filters blood", s(2): kidney, f(2) s(2): filters blood; "enzyme changes lactose", s(3) enzyme, f(3) s(3): changes lactose.
f(x): verb (self)	Does the phrase imply "SELF" as the object?	As in "the lizard walks", or "the bird flies", or the "protein unravels", the implication is that the action is performed by the system on itself.	"the lizard walks", f(1): walks (self); "the bird flies", f(2) flies (self); "the protein unravels" f(3) unravels (self)
RELATIONSHIPS (Functions, Systems)			

f(x-n)	Is the phrase a SUB-FUNCTION?	Sub-functions are necessary in order to achieve some higher level functional objective. For example, the "the lizards tail stabilizes the lizard, which is necessary for it to walk", contains the functions (1) "lizard walks" and (2) "tail stabilizes lizard". In this case, function (2) is a sub-function of function (1). The phrases "in order to", "so that", "due to" are usually indicative of sub-functions. Multiple sub-functions can be attributed to a single function, for example "in order for a lizard to walk, it must slap its feet alternatively on the water, while usings its tail to stabilize itself". A sub-function can also have sub-functions, for example, "the fiddler crab eats, by filtering sand with its mouth, which has setae that separate the sand from the organic material."	"gymnastic chalk absorbs perspiration in order to increase the friction between hand and rock" which is coded as f(1):increases friction, f(1-1): absorbs perspiration; "the fiddler crab eats by sucking in sand with its mouth which has setae that separate sand from organic material" is coded as f(1): eats (organic material), f(1-1): sucks in sand; f(1-1-1) separates sand. (solutions excluded)
--------	-------------------------------	--	--

f(x) [b, s, sd](n);	Is the function related to a SYSTEM?	<p>Functions are related to (1) an existing system, (2) a conjectured design, or (3) a problem. An existing system may be either biological or technological, whereas conjectured designs are (usually) technological. Sometimes functions are NOT related to specific solutions, but are stated as (unsolved) problems. For example "filtering water is a problem for humanity". While we know there exist solutions to the problem, in this case the function "filtering water" exists in isolation (for now), from any related solutions.</p>	<p>"gymnastic chalk absorbs perspiration" is coded more completely as s(1): gymnastic chalk, f(1) s(1): absorbs perspiration</p>
<p>SUB_TYPES (Maintenance, Prevent, Allow, Negation)</p>			
f(x, m)	Is the function a MAINTENANCE function? (sub-type, m)	<p>Functions that maintain a state are considered maintenance for example "the thermostat maintains temperature" is a maintenance function. Words like regulate, and control are also sometimes maintenance functions. Maintenance functions return a system to a desired state.</p>	<p>"moth maintains position", s(1): moth, f(1, m) s(1): maintains position</p>
f(x, p)	Is the function a PREVENTATIVE function? (sub-type p)	<p>Preventative functions keep a state OR a function from occurring, for example "armor prevents damage," "bad roads prevent transportation," or "the lock prevents opening". The opposite of PREVENT is ALLOW</p>	<p>"bad roads prevent transportation", s(1): bad roads, f(1,p) s(1): prevent transportation; "the lock prevents opening", s(2): lock, f(2,p) s(2) prevents opening</p>
f(x, a)	Is the function an ALLOW function? (sub-type, a)	<p>Allow functions enable a condition or a function to occur, for example "loose clothing allows movement", or "the filter allows water to pass through"</p>	<p>"loose clothing allows free movement", s(1): loose clothing, f(1, a) s(1): allows movement</p>

f(x, n)	Is the function a NEGATION function? (sub-type, n)	Negative functions are stated as NOT performing another function, for instance "LEDs do not balance colors", "this application does not produce light", "controlled burns do not remove spilled oil".	"LEDs do not balance color", s(1):LED, f(1,n) s(1): not balance color, "application does not produce light", sd(1):application, f(1, n) sd(1): not produce light
f(x, c)	Is the function a ACCOMPLISHMENT function? (sub-type, n)	By default, functions are accomplishment functions	

SUB_TYPES (AND, OR)

f(x, OR[n]), where N designates a shared group of mutually exclusive functions.

Is the sub-function one of many possible mutually exclusive sub-functions ?

Where there are multiple sub-function relationships for a given function there can be two types of relationship among the sub-functions: AND-type and OR-type relationships. The AND-type relationship is one that specifies that the related sub-functions must all be accomplished in order to achieve the super-function. The OR-type relationship specifies that one of the functions must be accomplished to achieve the super-function. AND-type is assumed unless OR-type is specified.

"We could move the vehical either using walking, rolling, or slithering. We need to choose one." f(1): move (self), f(1-1, OR[1]): walk (self), f(1-2, OR[1]): roll (self), f(1-3, OR[1]): slither (self).

f(x, AND[n]), where N designates a shared group of necessary functions.

Is the function one of a number of sub-functions that all have to be accomplished to accomplish the super-function?

The AND designation is assumed, unless it needs to be made explicit for clarity.

"in order to slither, we need to do several things: bend, straighten, and coordinate movements between sections" f(1-3): slither (self), f(1-3-1, AND[2]): walk (self), f(1-3-2, AND[2]): roll (self), f(1-3-3, AND[2]): slither (self).

What to watch out for.

Phrases may include actions taken by entities being described other than a "solution" or "design" entity. For example, actions taken by the designer, a researcher, a scientist, or by an instructor are not considered functions from the perspective of the design problem. In the sentence "scientists found that lizards fly" the phrase "scientists found" is not a relevant function, whereas "lizards fly" is (unless the design problem is about developing a tool to help scientists find lizards).

Words sometimes have to be reordered to make the function apparent. For example the phrase "I noted how remarkable the pattern was for packaging the overall large size of the flower relative to the bud" contains the function "pattern packages flower". [[The phrase "I noted how remarkable" is about the designer, and not of interest. The phrase "overall large size relative to the bud" expresses the size of the flower relative to some other state, which is a kind of specification.]]

The verb "uses" is usually not a function-verb. For instance "the lizard uses wings to fly" suggests the function "wings fly lizard" and not "lizard uses wings".

It is sometimes not clear if the design is one that is conjectured (the designers have in mind to propose) or existing. Often designers discuss other designers conjectured designs, for example "scientists at MIT are working on a robot that walks on water". Is this robot an existing solution or conjectured. For consistency, any system that is not conjectured as a design solution for the problem that the team is working on, is considered an EXISTING solution.

The word "ALLOW" is not always indicative of a function. For example, "cardboard allows for a lightweight structure," is the same as indicating cardboard has the property of being lightweight, which is a material property specification (see specification)

A negative function is often found in the context of DEFICIENCY or a BENEFIT of a solution. For phrases like "and the benefit of this system is that it won't create excess heat..." or "unfortunately the system can't balance itself..." the later half of each phrase is coded as the negative function, while the former half of each phrase is coded as the deficiency or benefit (with a relationship to the negative function).

Interestingly, designers rarely tend to state OR-type functions, it seems, somehow "reserving the right" to use any combination of functions as needed. A more formal "matrix chart" (like a morpho-matrix) of high-level requirements, and options for accomplishing each sub-level would generate [AND] functions at the high level and [OR] functions at the sub-level.

OPERATIONAL ENVIRONMENT

Code Form	Question	Criteria	Examples
	Does the phrase refer to an operating environment?	An operating environment refers to the locations, conditions, times and entities -- such as users, or other organisms/interactive systems -- the describe the surroundings of the device during normal operation.	"on a hill", oe(1, l): on a hill;" during a storm", oe(2, t): during a storm; "high temperatures", oe(3, c): high temperatures; "maintenance crews", oe(4, u): maintenance crews, "bacteria", oe(5, e): bacteria; "sharks", oe(6, e): sharks, "other motorized vehicles", oe(7, e): motorized vehicles

SUB-TYPES (Location, Condition, Time, User, Entity, Sub-system)

oe(x, l)	Is the phrase describing a LOCATION?	These are most common, and describe the places in which the system is intended to operate. The place may be somewhat generic "on earth", "at sea", or more specific, "in my apartment". These almost always contain a preposition phrases, such as "at", "in", "on", "near to", etc.	"on earth", oe(1, l): on earth; "at sea", oe (2, l): at sea, "in my apartment" oe(3, l): in my apartment
----------	--------------------------------------	--	--

oe(x, c)	Is the phrase describing a CONDITION?	These describe the conditions under which the system is intended to operate. These could be high/low end values, expected values, or ranges. Often times the values are QUALITATIVE and not QUANTITATIVE, for example "at high temperates" instead of "at temperatures of 500F degrees". Often times conditions occur in conjunction with a time - for example "while it is foggy", which includes both time and condition information, coded oe(1, c): foggy, oe(1-1, t) while. Conditions are also often linked to locations, so "in the mountains it is often foggy at dawn" would be coded as oe(1, l): in the mountains, oe (1-1, c) foggy, oe (1-1-1, t): at dawn.	"at high temperatures", oe(1, c): high temperatures, "at temperatures of 500F degrees", oe(2, c): 500F degrees, "wet and slippery", oe(3, c): wet, oe (4, c): slippery
----------	---------------------------------------	--	--

oe(x, t)	<p>Is the phrase describing a TIME period?</p>	<p>This describes the time during which the system must operate for example, "at night." Words like "when", "after", "while", "as" and "during" are often used to express a temporal environment. In the phrase, "the system remains viable after two years", "after two years" describes a time, coded oe(1, t): after two years. Time periods very often co-occur with conditions such as "when it is dark" which is coded oe(1, c): dark, oe (1-1, t): when. We code the time environment as a temporal sub-type of the condition.</p>	<p>"at night", oe(1, t): at night; "on rainy days", oe(2, c): rainy, oe(2-1, t): on rainy days; "for 3 hours a day", oe(3, t): for 3 hours a day</p>
oe(x, u)	<p>Is the phrase describing an intended USER of the system?</p>	<p>The phrase describes an intended user or class of users for the system, often including demographic data. For example, "the tool is for use by students, between the ages of 12 and 14", which we code as oe(1, u):students, oe(1-1, u): ages 12 to 14</p>	<p>"students", oe(1, u): students; "installation workers", oe(2, u):installation workers; "people in poor countries", oe(3, l): poor countries, oe(3-1, u): people in poor countries</p>
oe(x, e)	<p>Is the phrase describing a non-user ENTITY that could interact with the system?</p>	<p>The phrase describes an entity, often biological but sometimes technological, that interacts with the system. For example sharks that interact with shark nets, or bacteria that grow on medical devices. This could even include things like other similar systems, such as other vehicles on the road, or things like weapons being fired at the system.</p>	<p>"sharks", oe(1, e): sharks; "bacteria", oe (2, e), bacteria; "missles and land mines", oe(3, e):missles, oe(4, e): land mines.</p>
oe(x, s)	<p>Is the phrase describing a SYSTEM within which the system will work?</p>	<p>The phrase describes another system within which the system is intended to work or to connect to, for example "the new heat sink design could work inside a computer, or in HVAC systems" which is coded as oe(1, s): inside computer, oe(2, s): in HVAC systems.</p>	<p>"in a computer on the CPU", oe(1, s): in a computer, oe(2, s): on the CPU; "the system needs to work inside a human", oe(3, s): inside a human.</p>

		Operating environments often to provide additional details for FUNCTIONS, usually in the form of a preposition following the function. Functions that alter conditions, often have condition environments after, for example in the phrase "maintain heat at 100F", would be coded as f(1): maintain heat, oe(1, c) f(1): at 100F; likewise functions that alter location often have location environments following them, for example "the spider rolls itself to the bottom of the sand dune" would be coded b(1): spider, f(1) b(1): rolls (self), oe(1, l) f(1): bottom of the sand dune
oe(x) f(y)	Is the operating environment linked to a FUNCTION?	
oe(x) [b, s, sd](y)	Is the operating environment linked to a specific SOLUTION?	Operating environments are often associated with a particular solution that already exists and operates within that environment. For example, "the lotus plant is found in dirty conditions" would be coded as b(1): lotus, oe(1, c) b(1):dirty,

What to watch out for.

Often environments, users and times that are expressed are not relevant to solutions or design problems. In the sentence: "last year at Georgia Tech, scientists discovered proteins that keep cell from rupturing even when frozen to -100F," if the design problem is about keeping drugs viable in the cold, the terms "last year", "at Georgia Tech", and "scientists" are not operational environment factors, although -100F is.

Not all time-related phrases involve the operating environment. Phrases like "in the future" or "in the past" where a designer is prognosticating about future or past developments are usually not important elements of the operating environment. Remember there is a difference between systems used as analogies and systems that describe the environment within which the system must work

SPECIFICATIONS/CONSTRAINTS

Code Form	Question	Criteria	Examples
a(x)	Does the phrase establish a boundary or value concerning the manufacture or implementation of the actual designed artifact?	The phrase concerns constraints or specifications relating to the manufacture or final design of the created artifact itself. This includes material, shape, measurements, physical characteristics, components, component relationships, cost (to manufacture or operate), et al. This includes for example, pre-defined customer constraints. The phrase may also refer to an existing design or solution.	
SUB-TYPES (existing, limiting, enabling)			
a(x, l)	Is the phrase limiting choices or removing options? (sub-type limiting, l)	Limiting specifications/constraints are those which require a designer to use a smaller subset of design elements (see sub-types below) than would otherwise be available.	Limiting phrases use words like "must have", "should", "needs to use". For instance "we need to use non-toxic materials" is a phrase which limits the design to ONLY non-toxic materials which we would code a(1, l):non-toxic materials
a(x, e)	Is the phrase enabling choices or creating options? (sub-type enabling, e)	Enabling specifications/constraints offer new possibilities for design elements without enforcing their use.	Enabling phrases use words like "may have", "possibly", "could". For instance "we could make the design out of metal foam" is a phrase which enables a new potential design idea which we would code sd(1): new design, a(1, e) sd(1): metal foam
a(x, x)	Is the phrase concerning an existing design? (sub-type existing x)	Existing specifications/constraints discuss the specific properties of an existing design.	"The car jack uses a lever" would be coded as s(1): car jack, a(1, com) s(1): lever
SUB-TYPES (material, information, energy, time, shape, structure, structural relationship, cost)			

a(x, mat)	<p>Is the phrase concerning material, material property, or property value(s) of a material property of the artifact?</p>	<p>The phase is concerning the material of which one or more components of the design will be composed. The phrase could specify a material or range of materials, specify characteristics of the material or specific values for those characteristics.</p>	<p>"we should use steel", a(1, mat): steel; "we should use a material that is strong", a(1, mat): strong; "we should use a material with a tensile strength of 250MPa", a(1, mat): 250MPa.</p>
a(x, inf)	<p>Is the phrase concerning signals, signal types, or information or property/values of the same, which the artifact will use (internally)?</p>	<p>Information can be in the form of energetic signals, bits and bytes, or may be encoded in the physical structure of a thing.</p>	<p>"we can use feedback signals", a(1, inf):feedback signal; "the music is encoded as ridges in the record vinyl" a(2, inf): music, f(1): encode music, a(3, mat):record vinyl, a(4, shp) a(3) ridges;</p>
a(x, eng)	<p>Is the phrase concerning energy, properties or energy, or values for thoes ppoerties of energy which the artifact will use (internally)?</p>	<p>Energy can be found throughout a system in many forms; potential, kinetic, electrical, chemical, thermal, etc. The energy sub-type is used when a specified form of energy is discussed within the confines of the system.</p>	<p>"the current flows through the wires"; a(1, eng): current, f(1) a(1): flows (self), a(2, com): wires; "heat is transduced to electricity" a(3, eng): heat, f(1) a(3): transduced, a(4, eng): electricity.</p>
a(x, tim)	<p>Is the phrase concerning (non-operating environment, non-functional) timing such as manufacturing time, transit time, expected product lifecycle periods, degradation time for materials, etc.?</p>	<p>Phrases that include timeframes not related to the timing of the operation of the design -- such as how long manufacturing may take, how quickly the product must be installed, how long components or materials may last, etc.</p>	<p>"a mayfly only lives for one day"; b(1):mayfly, a(1, tim) b(1): lives for one day; "the manufacturing process takes months", a(2, tim): months to manufacturing; "the part must last 30 years", a(3, com): part, a(4, tim) a(3): last 30 years</p>

a(x, com)	Is the phrase concerning a component?	Phrases that include descriptions of specific parts of a solution or design, or groups of parts (e.g. "a leaf" or the "leaves" of a tree); generally components do not change during the operation of a system.	
a(x, pro)	Is the phrase concerning a property/value of the system in general?	Phrases that concern properties of the system as a whole or their values,	"the system must be lightweight"
a(x, shp)	Is the phrase concerning component shapes?	Any phrase discussing the shape of the components or of the solution/design itself.	
a(x, spc)	Is the phrase concerning spatial orientation of a component or sub-system?	This specifies the SPATIAL relationship or orientation between or among one or many components, systems, or sub-systems. This could either be with respect to an absolute frame of reference, with respect to other elements internal to the design, or between external (environmental) elements and elements of the solution/design (or the solution/design as a whole)	above, below, to the right of, etc.
a(x, rel)	Is the phrase concerning structural relationships among components or sub-systems?	This will require specifying which components are related by means of joints and contacts points.	
a(x, cst)	is the phrase concerning cost?	Usually in monetary terms, but this could also be in terms of any resource of concern. This may be an absolute or relative number e.g. "that will be very expensive" or "that will cost \$200".	

RELATIONSHIPS (specifications/constraints, solutions, operating environments, functions)

a(x) a(y)	Is the phrase related to another specification or constraint?	Specifications can often follow one from the other. A material may be specified, and then a range of properties and values for that material may be further specified. Components often are further specified in terms of shapes, relationships, and materials, etc.
a(x) s(y)	Is the phrase related to an existing solution or design?	Specifications, especially component specifications, are often related to specific existing solutions or proposed designs. Sometimes specifications arise because of a particular environment, for example a "water-proof materials" specification may arise as a result of an "underwater" operating environment.
a(x) oe(y)	Is the phrase related to an operational environment?	
a(x) f(y)	Is the phrase related to a function?	Some specifications, especially components, but also materials and others, enable or are associated with a particular function for the design. For example a "wing" component specification may arise as a result of the "flying" function.

What to watch out for.

Specifications may be confused with environments, where the phrase in question represents an entity that crosses the artifact/environment boundary. The exact wording of a phrase is critical in disambiguating between categories. See energy and information sub-types below for examples.

Specifications are very often related to solution designs (SD) or existing designs (S, B).

Note that "uses" rarely refers to a function e.g. "uses lever" is not a function, but rather a way of saying "a lever is a part of the system".

Stating that the device will be used *IN* an environment with signals available to it (e.g. "the device will be used in places where people will be calling out for help") is a characteristic of the environment and therefore an OE. Stating that the device will USE a specific kind of signal e.g. "the input to the device will be an audio signal generated by a call for help, as well as external noise from the environment which must be filtered."

Keep in mind the domain here is mechanical engineering. Computer science, computer engineers, and electrical engineers may have to redefine materials and information to correspond better with their notions of design.

Stating that the device will be used *IN* an environment with specific power characteristics (e.g. the device will be used in American markets with standard 120V power outlets), is a characteristic of the environment and therefore an OE (often OEs assume the specification). Stating that the device will USE a

specific kind of power (e.g. the input to the device will be 120V A/C) is a specification. The line between OE and Specification can be very blurry as the mention of the OE often implies the specification without stating it explicitly.

A spatial relationship may have two or more relationships between the two elements that are being related

A structural relationship will have two relationships (at least); the two elements (or more) that are being connected

When "time" is referred to in terms of a limited resource being "spent", something like manufacturing time may be construed as a kind of cost. For example "we don't have the time to manufacture the required number of widgets", or "its going to take 20 minuntes per unit to make those, versus 18 minutes for the other option", we see time being used as a measure of a limited resource. Generally if "cash" would equally fit the phrase, then it could be associated with cost.

PREPOSITIONAL PHRASES

Code Form	Question	Criteria	Examples
prep(x, f(n), b)	Is the phrase a prepositional phrase that adds more specificity or refines a function?	A prepositional phrase that comes at the end of a function and adds new information to the function, often times either indicating the operational environment or a substance/specification that the function is operating over.	"the pump moves water through the pipe," s(1): the pump; f(1) s(1): moves water; prep(1, f(1), spec(1)): through; spec(1, com):the pipe ; "the basilisk lizard walks on water," b(1): basilisk lizard; f(2) b(1): walks (self); prep (2, f(2), oe(1)): on; oe(1): water

Prepositional Sub-types (OE, Spec)

prep(x, f(n), oe)	Is the phrase a prepositional phrase that specifies an operational environment constraining or specifying where the function operates.		"ants pull food into the nest," b(1): ants; f(1) b(1): pull food; prep(1, f(1), oe(1)): into; oe(1): nest
prep(x, f(n), spec)	Is the phrase a prepositional phrase that specifies a component or material of the solution that the function operates on (or transforms something to)?		"the kidney changes waste into urine," b(1): kidney, f(1) b(1): changes waste, prep(1, f(1), spec(1)): into; spec(1,mat): urine

What to watch out for.

some prepositional phrases use short-hand, for example "in" may mean "inside of," "into," or "in order to". When used as "in order to" it is usually indicative of a sub-function, and not coded as a prepositional phrase.
 the prepositional phrases "when" is classified in operational environment - time; not as prepositional phrase.
 some prepositional phrases do not relate to a function; for example "the bacteria within the intestines" provides a relationship between bacteria and the intestines, and is not related to a function directly.

PERFORMANCE CRITERIA

Code Form	Question	Criteria	Examples
p(x)	Is the phrase related to performance or performance criteria?	Performance and performance criteria explain the <i>degree to which</i> or <i>way in which</i> an existing design performs or a new design must/is anticipated to perform its function in order to be judged a good solution. <i>Degree</i> usually specifies an absolute or relative evaluation metric (performance variable) which the existing design performs or the new design must achieve, for example "must achieve a velocity of 50mph," "must reach a depth of 500m," or "must be faster than existing designs". <i>Degree to which</i> is often underspecified both in terms of quantification, and in terms of to which aspect of the design the criteria applies. The phrase "must be a more efficient design" does not specify how much more efficient or what aspects must be more efficient. <i>Way in which</i> specifies a general characteristic of the way in which a function should be performed; for example the modifiers "passively" and "sustainably" are criteria for specifying qualitatively how the function is to be performed, without generating specific sub-functions, mechanisms, or structures, etc. Performance criteria modify functions.	"must achieve a velocity of 50mph", f(1): fly, p(1) f(1): velocity of 50mph; "must run faster than existing designs", s(1):existing designs, f(2): run; p(2) f(2) s(1): faster than; "must reach a depth of 500m" f(2):dive, oe(1): 500m depth, p(3) f(2) oe(1): reach 500m depth; "passively", p(4) f(all): passively; "sustainably", p(5) f(all): sustainably.

RELATIONSHIPS (Functions, Solutions)

p(x) f(n); p(x) f(1,2,3); p(x) f(all)	Is the performance criteria related to one or more functions?	The performance criteria is related to a function, a set of functions, or "all" functions (otherwise it is a specification). Problem descriptions can often be vague with respect to which function the criteria is being applied to. For example the criteria "must be faster than existing robots" uses the term "be" when the design may be referring to its mode of movement, its recharge rate, or its ability to make decisions. The coder must draw on context to make the best guess.	
---	---	---	--

p(x) f(m) s(n);	Is the performance criteria related to an existing solution?	The criteria explicitly refers to an existing solution. In the example "must be faster than existing robots", the performance criteria is citing the performance of "existing robots" which is an existing solution. The criteria could be functionally non-specific such as "must be better than existing robots", in which case we use the "all" designation.	
------------------	--	---	--

SUB-TYPES (Absolute, Actual, Relative)

p(x, s) f(n);	Is the performance metric explicitly stated as a specific value or range? (specific, sub-type s)	The phrase states the specific value or range of the performance criteria.
p(x, a) f(n);	If related to an existing solution - is the phrase stating the <i>actual</i> performance of the solution? (actual, sub-type a)	The phrase simply states the performance of an existing solution, for example "solution(n) can reach speeds of 120mph". This is not stating that the new design performance must reach these speeds, but rather stating how fast an existing solution is. This is a special case of the absolute criteria.
p(x, r) f(n) s(n);	Is the performance criteria a relative measurements to an existing benchmark or solution e.g. is the criteria comparative. (relative, sub-type r)	The phrase uses comparative terms such are "quieter than solution(n)", "faster than solution(n)", "more efficient than solution(n)", without explicitly stating the performance of solution(n). In this case, there will always be a solution mentioned designated by the " s(n)" coding.

BENEFITS and DEFICIENCIES

Code Form	Question	Criteria	Examples
ben(x), def(x)	Is the phrase pointing out a relatively positive or negative association (a value judgement) with either an existing product or an expectation?	Benefits/deficiencies can relate to any other element of an existing solution or proposed design, highlighting a favorable or unfavorable aspect of that element. They must always refer to another element and specify a value judgement about that element "good, better, wonderful, beneficial," etc. OR "bad, worse, terrible, detrimental, is a problem" etc.	

What to watch out for.

The benefit or deficiency may be a single keyword like that is embedded in a longer phrase that provides the element to which the ben/def is referring. The phrase - "x is a problem" can be indicative of a deficiency, but not always. When the problem is with respect to A SOLUTION (DESIGN, BIOLOGICAL, PROPOSED) then it is a deficiency. When the problem is META-level, then it is NOT a deficiency (of the solution). For example: "one problem is there is not enough known about how the biological system works" is NOT a deficiency of the SYSTEM, but of our knowledge of the system, which is META.

		LIFECYCLE STAGE (not used)	
Code Form	Question	Criteria	Examples
		SUB-TYPES (manufacturing, installation, transport, operation, maintenance, recycling)	
l(x, mn)	Is the phrase concerning manufacturing of the artifact? (sub-type manufacturing, mn)		
l(x, in)	Is the phrase concerning installation of the artifact? (sub-type installation, in)		
l(x, tr)	Is the phrase concerning (non-operational) transport of the artifact? (sub-type transport, tr)		
l(x, op)	Is the phrase concerning the normal operation? (sub-type operation, op)		
l(x, mt)	Is the phrase concerning maintenance of the artifact? (sub-type maintenance, in)		
l(x, re)	Is the phrase concerning recycling or disposal of the artifact? (sub-type recycling, re)		

OPEN QUESTIONS (not used)

Code Form	Question	Criteria	Examples
Function SUB_TYPES (Contingency)			
f(x, follows) f(y)	Is the function temporally contingent on the execution of another function?	The function occurs after a different (non-sub) function.	
f(x, requires) f(y)	Is the function temporally contingent on the execution of another function?	The function requires or is enabled by the execution of a different (non-sub) function.	
Solution SUB_TYPES (Component)			
[b, s, sd](x-n, c)	Is the solution a SUB-COMPONENT of another solution? (sub-type c)	<p>A sub-component has a "part-of" relationship with another system, but consists of a single entity. A pipe is a sub-component of a heat exchanger, coded as s(1): heat exchanger, s(1-1,c): pipe</p> <p>A sub-component should be coded as a specification (sub-type component)</p>	"the shrimp-gun has a hammer, a plug, and a tension band"; sd(1): shrimp-gun, sd(1-1, c): hammer, sd(1-2, c): plug, sd(1-3, c): tension band.

What to watch out for.

A complex sub-component with many parts should be coded as a sub-system.

APPENDIX G: Example design document (2009) and SR.BID coding

This appendix provides an example design document from the 2009 problem definition assignment, and the accompanying SR.BID encoding.

Problem definition document text:

I think that our charette went over pretty well and that we got some wonderful feedback that will help give us new insights and improve on our project. A problem statement is currently to improve solar design for structural uses by increasing efficiency, introducing sustainable materials, and making it dynamic. I think that our 5 analogies of the lily pad, diatoms, negative feedback, snail shells, and photosynthesis provided a wide range of possibilities to accomplish our current mission statement, but further decomposition is needed.

Solar panels would be more viable as an energy source if they can become better at conserving energy. This involves them becoming more efficient and would ultimately be more beneficial if they used more sustainable materials. Efficiency can be improved by making these structures more dynamic and respond to the environment. Another aspect of conserving energy is to make the transmission of energy more conserved, so that the amount of energy generated in a solar panel is the same transmitted to the energy receptors. Dealing with the sun, it is also important in energy conservation to retain energy when it is not sunny, or to initiate another process that will generate energy.

Focusing on making solar panels more dynamic, this can be done by having them clean themselves, respond to the environment, and be able to dissipate their own heat. Solar panels ultimately mimic plants, which respond to the environment and are fairly self sustaining. I think this is a big gap between the static and fragile solar panels that we have so far engineered. So far, most solar panels are set up on the grid basis, acting together, especially when moving to the sun, rather than as an individual.

Continuing off that tangent, I think it would be interesting to have an individual solar panel that can stand alone and still function. The snail shell structure is standalone and has the ability to passively dissipate heat by using the heat gradient, so that it is cooler within the shell than the outside air and ground. This would be helpful for allowing the interior of a structure with solar panels remain cool.

Currently solar panels are rigid and typically pretty sensitive. A flexible solar panel would be more resistant to changes in the environment and, depending on the materials, hardier and easier to maintain. They would be more moldable to varying structures and more portable.

A problem with rigid solar panels is that they can become dirty, which reduces the efficiency of light absorption. Introducing a nanostructure surface, like lily pads, would allow for a surface that can clean itself. This idea can also be applied to improved energy trapping by a new surface to absorb light.

This is just my train of thought on improving solar design for structures. I feel that narrowing down our problem will then help us in finding complimentary analogous solutions. I think the structure is very important for these improvements on solar panels, both current, and potential constructions.

The text is transcribed into a spreadsheet, and decomposed into individual concepts. Each concept is then encoded in terms of a primary encoding, and a relationship encoding. In rare instances, more than one relationship may be implied by the concept, such as when “make more efficient” is applied to a set of functions, instead of just one. Comments are logged at the time of encoding to capture subjectivity in the interpretation.

The following provides a encoding of the first two paragraphs of the text.

Table G-1 SR.BID encoding of two paragraphs of text

	Statement	Notes	Comment	Primary Concept	Relationship
1	I think that our charette went over pretty well and that we got some wonderful feedback that will help us give new insights and improve on our project.		Comment		
2	A problem statement is currently		Comment		
	to improve	a requirement on one or more of the functions F(any) t.b.d., of S(0)		perf(1): improve on (existing)	s(0): solar panel
3	solar design	the design of a solar panel, an existing solution S(0)		s(0): solar panel	
4	for structural uses	operational environment of S(0) - solar panels on stationary structures (e.g. roofs, ground, etc)		oe(1, 1): on statonary structures	
5	by increasing efficiency	is a comparative requirement on improving solar design		perf(2): more efficiently	s(0): solar panel
6	introducing sustainable materials			spec(1, mat): sustainable materials	s(0): solar panel
7	and making it dynamic.	it = the system, new sub of F(1)		f(1-1): make (self) dynamic	s(0): solar panel
8	I think that our five analogies of the		Comment		
9	lily pad,	specific solution		b(1): lily pad	
10	dynamic feedback,	general pattern		b(2): dynamic feedback	

Table G-1 continued.

11	snail shells,	specific solution		b(3): snail shells	
12	and photosynthesis	general process		b(4): phtotsynthesis	
13	provided a wide range of possibilities to accomplish our current mission statement but further decomposition is necessary.		Comment		
14	Solar panels would be more viable as an energy source	amplification of F(1)		f(1): generate solar energy	s(0): solar panel
15	if they can become better at conserving energy	new sub of F(1)		f(1-2): conserve energy	s(0): solar panel
16	this involves them becoming more efficient			perf(2): more efficiently	s(0): solar panel
17	and would ultimately be more beneficial if the used more sustainable materials.			spec(1, mat): sustainable materials	s(0): solar panel
18	efficiency can be improved by making these structures more dynamic			f(1-1): make (self) dynamic	s(0): solar panel
19	and respond to the environment	new sub of F(1-1), responding to the environment is a way of making the system more dynamic		f(1-1-1): respond to environment	s(0): solar panel
20	another aspect of conserving energy	the word "another" suggests a connection between conserving energy F(1-2) and efficiency F(1-1) in the previous statement		f(1-2): conserve energy	s(0): solar panel
21	is to make the transmission of energy	new sub of F(1-2)		f(1-2-1): transmit energy	

Table G-1 continued

22	more conserved	is a comparative requirement on the transmission of energy		perf(3): more conserved	f(1-2-1): transmit energy
23	so that the amount of energy generated in a solar panel	amplification of F(1)		f(1): generate solar energy	s(0): solar panel
24	is the same transmitted	amplifies the comparative requirement on the transmission of energy		perf(3): more conserved	f(1-2-1): transmit energy
25	to the energy receptors.	in the operational environment there are energy receptors		oe(2, u): receivers of energy	
26	dealing with the sun	the sun is part of the operational environment		oe(3, c): sun	
27	it is also important in energy conservation	related to F(1-2)		f(1-2): conserve energy	s(0): solar panel
28	to retain energy	new sub-function of F(1-2)		f(1-2-2): retain energy	
29	when it is not sunny	condition constraint: sun availability		oe(4, c): not sunny	f(1-2-2): retain energy, f(2): initiate another process
30	or to initiate another process	a function parallel to F(1),		f(2): initiate another process	
31	that will generate energy.	new super-function of F(1) generate solar and F(2) generate non-solar energy	super to f(1)	fs(1): generate energy	

APPENDIX H: Definition of Transformations

In general, problem transformation can be considered in three broad categories: conceptual addition, conceptual prioritization (including deletion), conceptual organization, and conceptual shifting. The following provides definitions for a partial taxonomy of observed transformations organized by addition, prioritization, and organization. Definitions are supported by both hypothetical examples, and one or more observed design instances in which they occur.

H.1. Conceptual Addition

Conceptual addition occurs when any new concept is added to an existing problem formulation, and is frequently added with an associated relationship. Concepts that are added without an existing relationship are perhaps related in the mind of the designer, but the relationship is not articulated or obvious. Conceptual addition occurs in a several regular patterns, depending on the nature of the relationships between existing and new concepts, and the order in which the addition occurs. The following patterns have been observed: Conceptual Refinement, Associative Addition, Abstraction Shifting, Induced Abstraction, Decomposition, or Disconnected.

H.1.1. Conceptual Refinement

Conceptual refinement occurs when a new concept is added such that it creates greater specificity for an existing concept of a different type. This may occur in many different conceptual combinations: *performance criteria* nearly always add additional specificity to a *function* (to reduce emissions by 98%); *operational environments* may

further specify a *function* (e.g. to walk *on water*, to fly *in turbulent air*) or a *solution* (e.g. spiders *in the desert*, plants *living in salt water*).

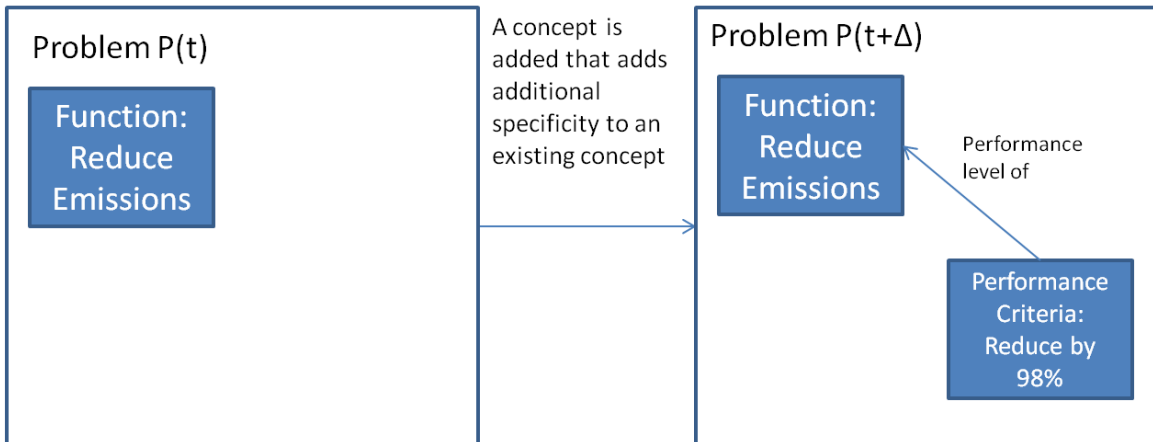


Figure H-1. Conceptual refinement transformation.

H.1.2. Associative Addition

Associative addition occurs when new concepts are added with a (non-refining) associative relationship with an existing concept of another type. For example “considering the desert, I know dessert snails live there and they have to keep cool,” starting with the *operational environment* the “desert,” new connections are formed with the *solution* “snail” and the *function* “keep cool”. This forms a heterogeneous association of new concepts associated with the original desert *operational environment*.

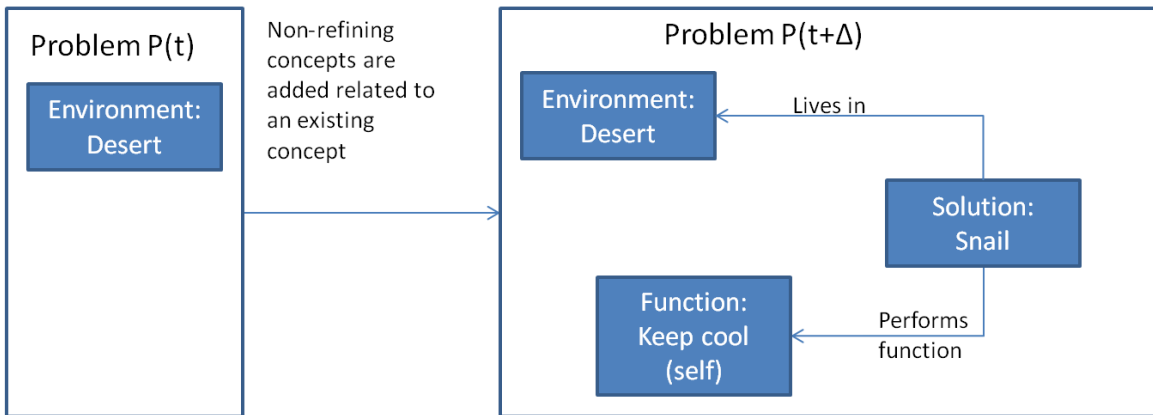


Figure H-2. Associative addition transformation.

H.1.3. Abstraction Shifting

Abstraction shifting occurs when a single parent or child node is added to a base concept. Abstraction shifting may occur by shifting up (zooming-out) in which a single parent node is added, or by shifting down (zooming-in) in which a single child node is added. Because the final state of the graph in abstraction shifting looks the same, identifying “shifting up” versus “shifting down” requires knowledge of the temporal ordering of the creation of the problem structure. *(research process comment: In the case of a text documents or statements of designer reflection temporal ordering may be difficult to infer. *

H.1.4. Shifting-Up

Shifting-up, or *zooming out*, occurs when a higher-level node of the same type is added to a single concept, for example for the *solution* category when a designer shifts from considering “pine trees” to considering “coniferous trees”. Likewise, a design may

abstract from specific *component* instance “e.g. the two-layer material of the pine cone scale” to a higher-level description of the *component type* “e.g. a bi-layer material”. It appears that in zooming-out, knowledge of the higher-order type category is known a priori. I distinguish shifting-up from induced abstraction (below), based on the number of nodes to which the higher-level node is added at the time at which the addition occurs.

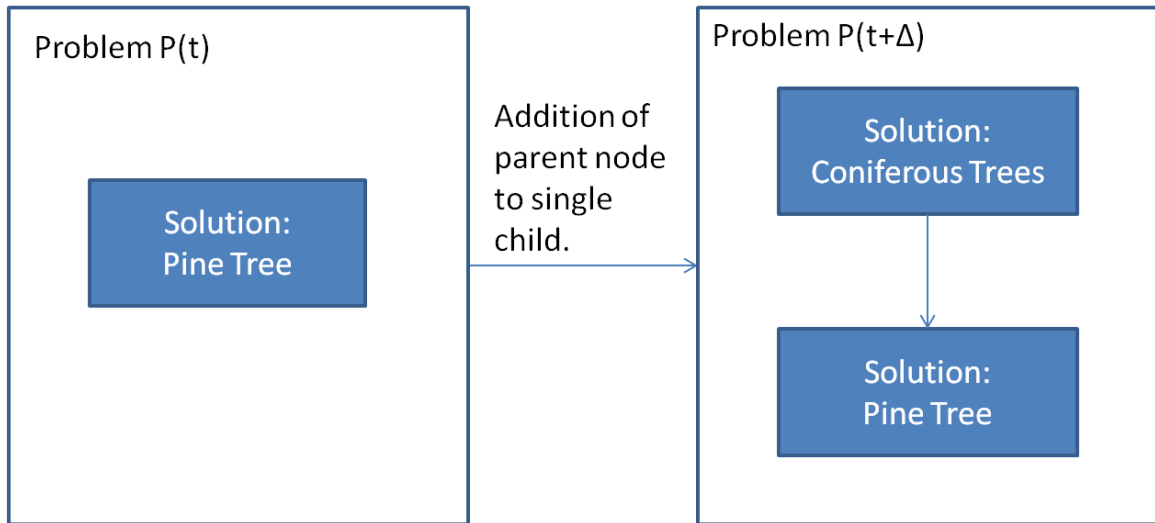


Figure H-3. Shifting-up transformation.

H.1.5. Shifting-Down

Shifting-Down or *zooming-in* occurs when a lower-level node of the same type is added to a single concept, for example for the *function* “generate power” the addition of the single *sub-function* “generate electricity” indicates a shift to a lower level of abstraction. In this case, generating electricity is method by which generate power may be achieved. Similarly, for an *operational environment* of “power plants”, the addition of the concept “coal-fired power plants” increases the level of specificity of the environment to one specific sub-type of power plant.

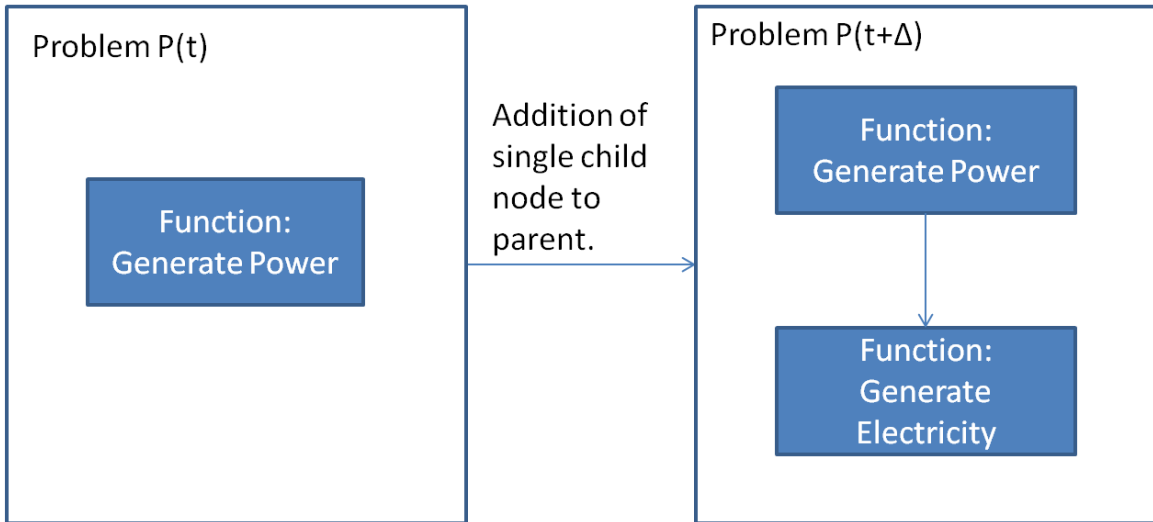


Figure H-4. Shifting-down transformation.

H.1.6. Induced Abstraction

Induced abstraction occurs when multiple instance of a conceptual category gives rise to a higher level super-category of the same type. Multiple-instance induced abstraction requires a category is induced from more than one instance. For example, the both instance of the *solution* Namibian beetle and the instance of the *solution* lotus leaf have *components* with super-hydrophobic surfaces to generate the *function* “cause super-hydrophobic effect.” A new *solution* super-category can be induced “super-hydrophobic solutions”, that includes both Namibian beetle and the lotus leaf as child nodes. This category is induced based on a common structure and function at one level (the super-hydrophobic property of the surface) despite the fact that the Namibian beetle uses the structure and function to collect water, while the lotus leaf uses the structure and function to maintain a clean surface. While the abstraction is created over the *solution* category, the features in common are a *component* and a *function*.

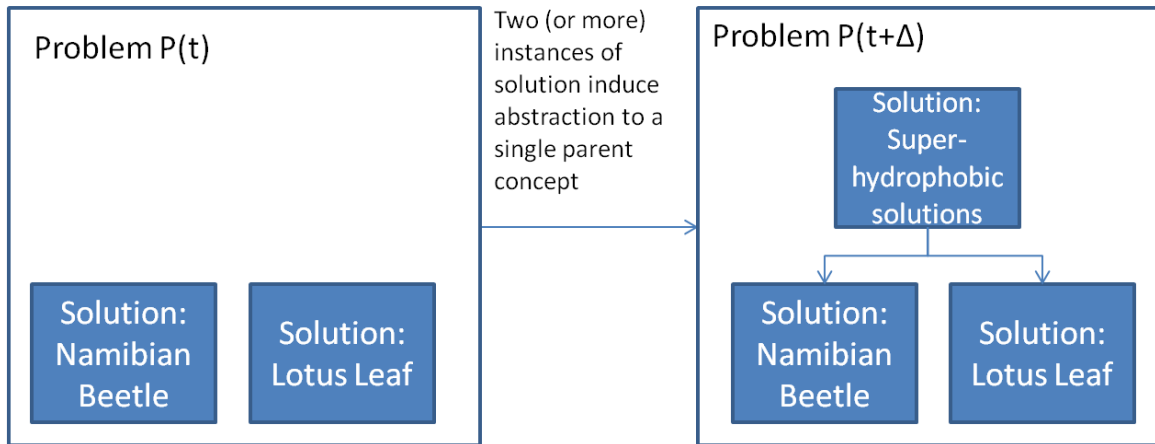


Figure H-5. Induced abstraction transformation.

H.1.7. Decomposition

Decomposition occurs when two or more child nodes are added to a parent concept, such that they divide the parent node into one or more constituent concepts. Decomposition appears to occur across all conceptual categories using nearly any divisible property. Any given decompositions will imply either a *conjunctive decomposition* or *disjunctive decomposition* with respect to the problem formulation. Thus for some concept C the decomposition $\{C(1), C(2), C(3)\}$ implies one of two things; in order to fulfill C for the *conjunctive* form, the solution must fulfill $C(1)$ AND $C(2)$ AND $C(3)$, or for the *disjunctive* form the solution may perform $C(1)$ OR $C(2)$ OR $C(3)$.

Decomposition may occur when multiple “type-of” instances are added to a single parent node (e.g. we can generate power by generating electricity, combusting chemicals, catching the wind, gathering sunlight, or tapping into heat sources). Taxonomic decomposition may occur across operational environments (e.g. we can increase living

space by expanding into the air, onto the water, or underground), as a temporal sequence of functions (e.g. in order to fly, we need to take off, remain aloft, and then land), or over different performance criteria (e.g. move stealthily at fast speeds, and move stealthily at slow speeds).

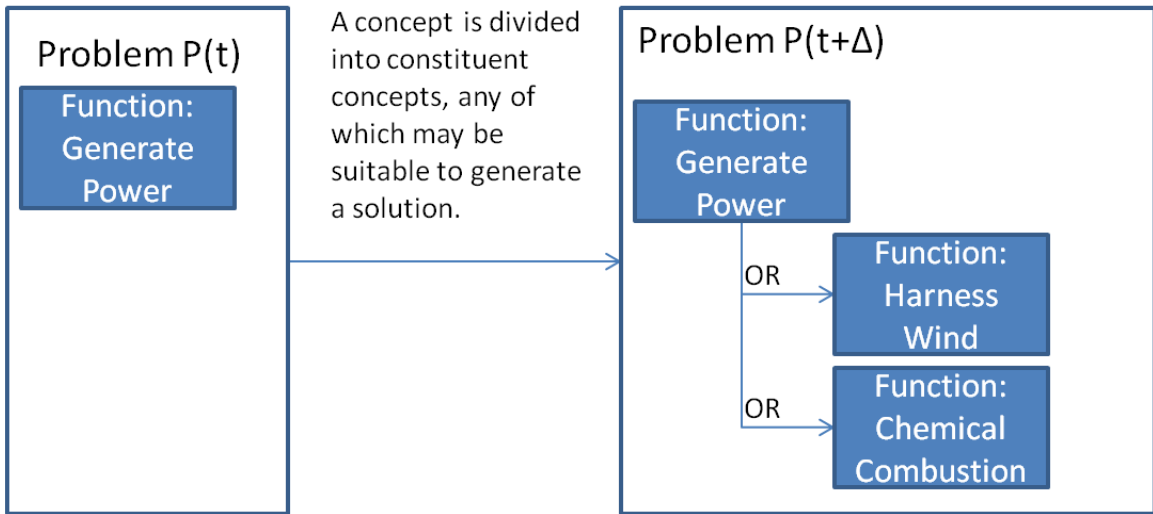


Figure H-6. OR-type decomposition transformation.

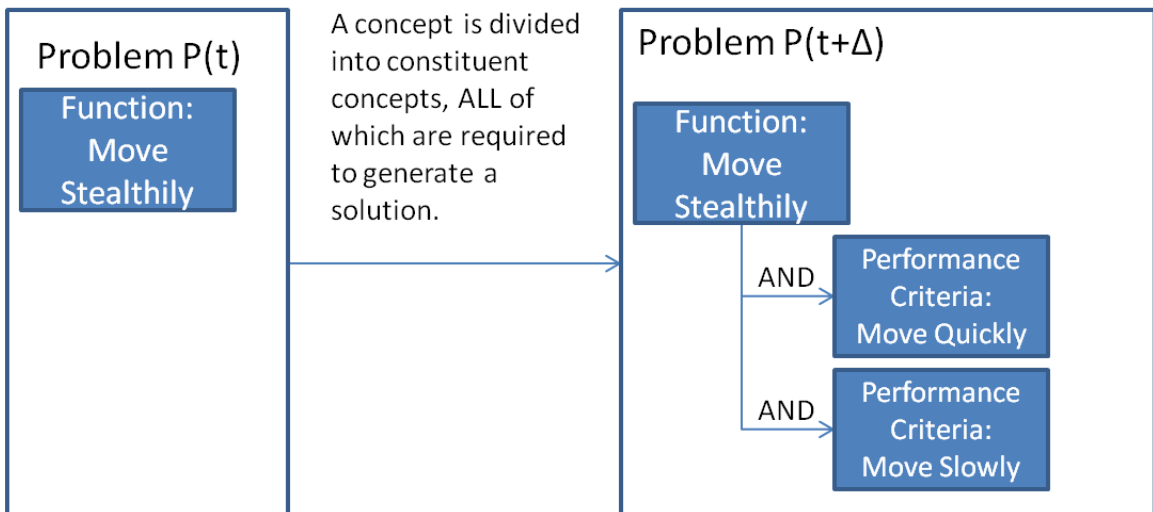


Figure H-7. AND-type decomposition transformation

H.1.8. Disconnected Addition

Disconnected addition occurs when a single concept is added to a problem formulation which is not immediately and directly connected to other concepts in the formulation (an isolated vertex). When student designers articulate design problems, connections between concepts are often tacit but recognizable. A designer need not explicitly mention that a pine cone is attached to a pine tree for the reader to understand the relationship between the two. Nonetheless, there are instances in which concepts are included during problem articulation for which relationships cannot be inferred. Most frequently occurring in this category are operational environment factors such as users and locations, which may be included in early problems formulations, but frequently remain disconnected from other concepts until and unless they become critical (especially problematic) aspects in the design (for instance due to discovery during evaluation).

H.1.9. Identifying differences between decomposition and abstraction

Abstraction is identified by the occurrence of a super-category relationship after the discussion of the instance from which the abstraction occurs. Although graphically, abstraction and decomposition may appear similar (a parent node with one or more child nodes), abstraction may be said to occur if the parent node appears (a) subsequent to the initial dialogue about the base concept, and EITHER (b) there appears to be one or more intervening lines of reasoning between the initial dialogue and the occurrence of the parent node OR (c) the designer articulates the process of abstraction (e.g. “if we think about this at a higher level”). It follows that decomposition occurs (a) when the parent node is appears first, and either (b) child nodes appear subsequent identified

H.2. Concept Suppression, Deletion and Reemergence

Frequently concepts “disappear” from one problem articulation to the next, sometimes to reappear later, while at other times not. The omission of a concept from future problem articulations is characterized differently than the intentional removal of a concept.

H.2.1. Concept Suppression

I define “concept suppression” as when a concept appears in one problem articulation, but (a) does not appear in the immediately subsequent articulation AND (b) in the previous iteration the concept held a relationship with a concept that DOES appear in the immediately subsequent articulation. Suppression implies that the concept is no longer the focus of attention during the problem articulation, and thus may be omitted as part of the active discussion, but since there exists an active concept to which it was related it may remain a relevant, though tacit, component of the problem model. Concept suppression allows for a kind of background connectivity to remain, accounting for the frequent reoccurrence of concepts in later iterations.

H.2.2. Concept Deletion

I define “concept suppression” as when a concept appears in one problem articulation, but (a) does not appear in the immediately subsequent articulation AND (b) in the previous iteration the concept DID NOT HOLD a relationship with a concept that appears in the immediately subsequent articulation. Concepts that bear no connection to existing concepts are often deleted in future iterations *disconnected deletion*. Concepts that form a related sub-graph that are removed from one iteration to the next are said to be a *cluster deletion*. Cluster deletions often occur when a partitioned sub-problem is

removed from consideration.

H.2.3. Concept Reemergence

A concept that was suppressed or deleted in a previous articulation, but appears in a future articulation is said to have reemerged. The reemergence may occur in relation to the concept to which it was previously related (*related reemergence*), or may occur in relation to a new concept (*novel reemergence*).

H.3. Connecting

H.3.1. Partitioning and Decoupling

Problem articulations that form independent sub-graphs with multiple conceptual types are said to be partitioned problems. Independent sub-graphs with one or more intermediate connecting nodes can said to be partitioned problems. Independent sub-graphs with no intermediate connecting nodes can be viewed as decoupled problems; that is there appears to be no relationship (articulated or obvious) between the two conceptual clusters. This may occur early in design when students are (a) considering multiple potential problems for further development (b) not aware of the connectivity between partitioned sub-problems.

H.3.2. Swapping and Incrementing

Periodically, a concept will be swapped with an existing concept. A concept is said to be swapped when (a) a new concept appears with all of the same relationships as the original concept, and (b) the concept itself bears no relationship with the original, except in kind (e.g. in trying to modify the efficiency of a solar panel to collect light, the

function “clean (self)” is replaced with the *function* “change color”, where the functions “change color” and “clean (self)” have the same connections to *solution* solar panel, *function* modify light collection, and *performance criteria* improve efficiency).

Incrementing is similar to swapping, except where swapping entails the creating of a new concept different than the first, incrementing entails a partial manipulation of the content of the existing concept. Changing existing performance criteria is an obvious example of incrementing. Less obvious might be shifting the object of a function for example, *regulate heat* changes to *regulate temperature*. In this case “temperature” is a broader reframing of the original “heat.”

APPENDIX I: 2012 End of Semester Survey

Fall 2012 Survey in Biologically Inspired Design

Name:

Major/Minor:

Year in school:

Please answer the following question using complete sentences.

Thinking about your final design problem, what do you think were the most challenging aspects of realizing the conceptual design?

Rank the following skills from one to five in order of importance in biologically inspired design. Give a Rank of 1 to the most important skill, a Rank of 2 to the second most important skill, etc. Use each Rank only once:

RANK

_____ Finding relevant biological sources of inspiration

_____ Understanding the underlying mechanisms of biological sources of inspiration

_____ Making correct analogies between biological sources of inspiration and design problems

_____ Defining design problems sufficiently and correctly

_____ Applying mechanisms from biology to the design problem correctly

What other skills do you think are critical for successful biologically inspired design?

- .
- .
- .
- .

On a scale of one to five, where 5 is VERY IMPORTANT, and 1 is NOT AT ALL IMPORTANT how important are the following aspects of a design problem.

RANK

_____Operational environment

_____Material constraints

_____Manufacturing constraints

_____Function

_____Performance criteria

_____Cost

What other aspects do you think are important for defining a problem in biologically inspired design?

-
-
-
-

On a scale of one to five, where 5 is VERY HELPFUL, and 1 is NOT AT ALL HELPFUL how well did SR.BID help you:

_____ Define your problem in a constructive way

_____ Communicate about your problem to others

_____ Define biological solutions in a constructive way

_____ Communicate about biological solutions to others

_____ Understand the accuracy of your analogy

_____ Communicate why you analogy was good/bad

Is there anything you would change about the use of SR.BID in class?

APPENDIX J: Evaluation of open-ended 2012 survey results

The analysis of open-ended questions was conducted using a category/sub-category coding scheme for the bullet-list/phrases, which was developed as the data presented itself. A separate category/sub-category scheme was developed for each question, although some cross-over did occur. Each comment was assigned to only one category/sub-category. For example, the sample provided earlier from the first question was coded as follows:

Table J-1. Sample survey answers and encoding.

Sample Answers	Category	Sub-category
Being able to move away from sunk cost	Process	Sunk cost
Bio-inspired materials are hard to assess for marketable/mass-produced product	Material	Production
Common sense check showed complete redesign required	Justification	Early Quantification
Feasibility and interaction analysis	Justification	Feasibility
Find the biological inspiration	Process	Search for Bio
Focusing on biological solutions vs technological and quantificaion	Process	Bio vs Eng
Justifying principle selected	Justification	Principle
Manufacturing feasibility	Manufacturing	Feasibility
Material selection	Material	Selection
Materials - developing a composite	Material	Development

The number of comments in each categories and sub-category were tallied to identify overall trends in responses. Table J-2 shows a breakdown of the number of comments

with respect to particular categories for the first question regarding which aspects of biologically inspired design are most challenging. Justification refers to those aspects which assess the feasibility or quantification of design choices. Materials and manufacturing is self-explanatory. Process comments either focus on a task within the design process, or discuss the design process as a whole. Team comments have to do with communication and consensus building. Sub-categories with less than one comment are omitted.

Table J-2. Example survey comments and count by category.

Category: Sub-Category	Example Comments	Count
Justification	“Feasibility and interaction analysis” “Quantification of important parts”	11
Materials & Manufacturing	“Materials selection” “Developing a composite”	9
Process: Application of Bio	“Applying the bio mechanism to the problem” “Staying True to the bio mechanism”	2
Process: Design Process	“Understanding the design process” “Did not understand the science of design”	2
Process: Inspiration v Copy	“Balance between bio-inspired and bio-mimicry” “Tried to mimic nature too much”	2
Process: Problem Decomp	“Problem decomposition to create a complete solution” “Problem decomposition”	3
Process: Search	“Finding the biological source of inspiration” “Finding the inspiration”	3
Team	“Conveying principles in a quick and easy manner” “Team agreement”	4

This table shows a bias toward activities that help to realize a design (quantification, material and manufacturing), versus conceptualization of the design. There are several productive interpretations of this data. One interpretation suggests that in fact justification and materials and manufacturing skills are the most challenging for BID, and that more time needs to be devoted to instruction on how to realize conceptual designs in BID. Another way to interpret the result data, is that the skills that are the focus in class, for

example search, analogy-making, problem definition, are being supported well enough in class that they are no longer considered difficult. Another related interpretation is that this could also be a reflection of grading and feedback -- it could be that students are focused on those aspects of their design on which they receive the lowest scores or most criticism.

The second question related to skills that are deemed critical for biologically inspired design. The following sub-categories all garnered more than one response. This data again reinforces the importance of understanding biological systems (see Table J-3) and the challenge of “hard” engineering

Table J-3. Critical BID skills and number of times cited in survey answers.

Skill	Number of Citations
General: Creativity	4
Knowledge: Biological	8
Knowledge: Engineering	7
Process: Application of Biological System	8
Process: Problem Specification	3
Team: Communication	6

The third open-ended question seeks to understand what additional concepts would be useful for specifying a problem. Feasibility suggests that the problem needs to be couched relative to the ability to solve the problem. Comments regarding market focus on market size, sales potential and the relevance of the selected problem to society. Two comments were with respect to couching the problem with respect to existing solutions. Note that SR.BID accounts for this in two ways, through *relative performance criteria* and through *benefits and deficiencies* (which were not included in the four-box method). Several comments were made suggesting that the four-box method of problem

formulation was redundant with SR.BID (which it is based on).¹⁹

Table J-4. Useful concepts for problem specification.

Problem Aspects	Number of Citations
Feasibility	5
Market	8
Redundant with SR.BID	3
Existing Solutions	2

The final question looks for improvement to the use of SR.BID in class, and seeks open, candid feedback. The main take away from this input is that some students felt that SR.BID was too constraining, that it should be an optional tool, and several students felt that more examples would be beneficial.²⁰

Table J-5. Opportunities for improving SR.BID.

SR.BID Improvements	Number of Citations
Too Constraining	4
More examples	4
Non-mandatory	2

¹⁹ According to conversations with the instructors, this redundancy is being addressed in the 2013 iteration of the class (four-box, and SR.BID are combined into a single conceptual framework, using the T-chart).

²⁰ Addressed in the 2013 iteration of the class, more examples of SR.BID and the four-box method are being made available to students.

APPENDIX K: ERR Diagram for the SR.BID Web Application

The figure K-1 provides a depiction of the entity-relationship diagram associated with the SR.BID Web application. The central organizing construct for the application is the PROJECT table. A project will be associated with one or more specific users. Each project can also be associated with elements of the four-box (or SR.BID) specification, each of which are likewise represented with their own table (function, operational environment, (OperationalEnv) performance criteria (Performance), and specification/constrain (ArtifactSpec). A project will have a description and optionally some ExternalFiles (images, pdf files, etc) associated with it.

The SOLUTION table is intended to capture biological source systems; because the information captured for these systems is the same as much of the captured for the PROJECT information (e.g. descriptions, four-box models, associated images and files), the solution reuses the “PROJECT” table to recreate the necessary data structures...this seems a little counter-intuitive and is in fact a little clunky. This is the reason that in the PROJECT table there is a Boolean field to designate if the PROJECT is actually a SOLUTION. A solution may also be associated with SBF-like entities, such as mechanisms, principles and components. The USER and Group Affiliation tables enable project and solutions to be associated with particular individuals and groups. The TRANSACTION table enables all adds/edits/deletes to associated PROJECT and SOLUTION content to be easily tracked by user and time of transaction.

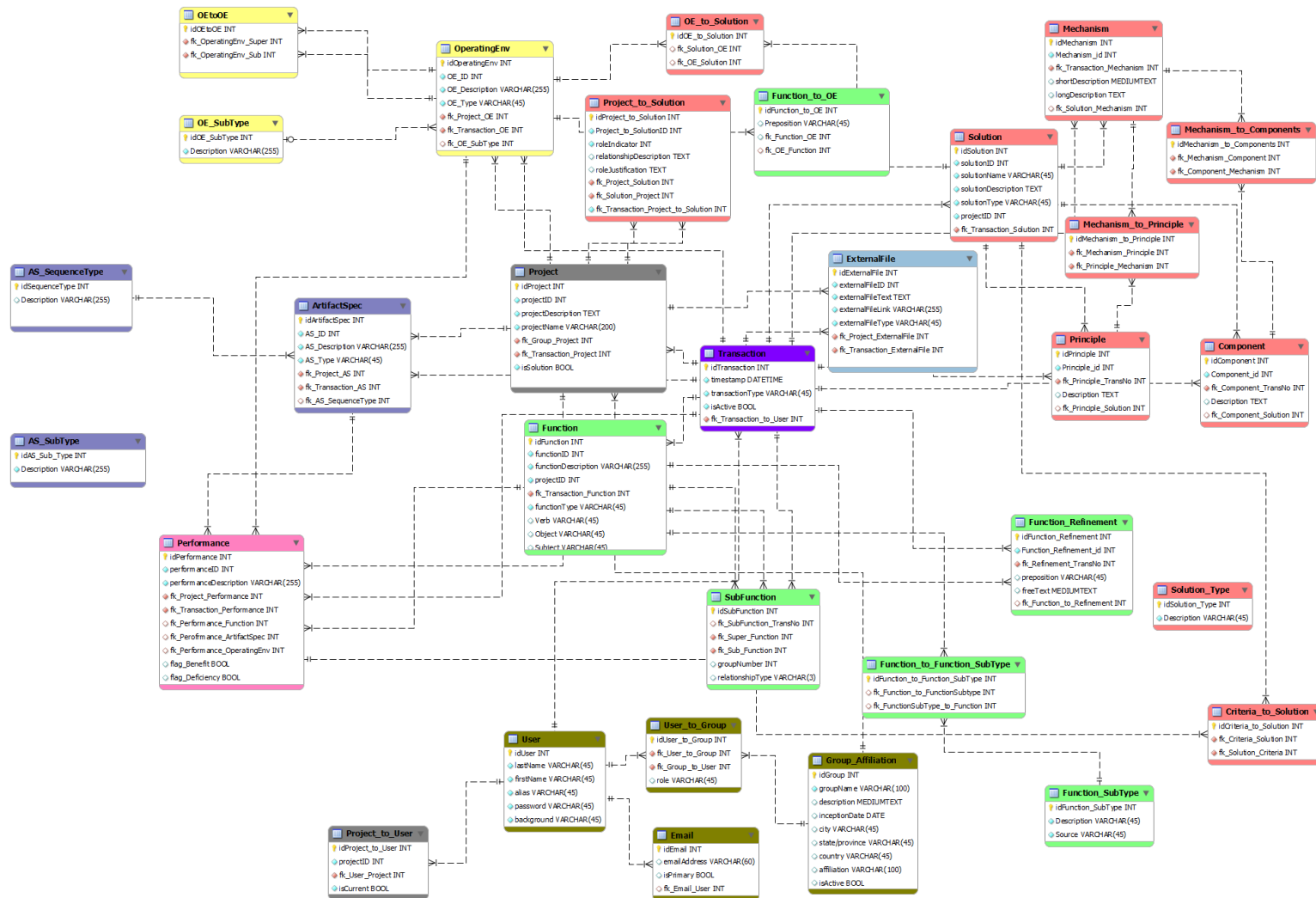


Figure K-1. Entity relationship diagram of problem-solution memory model.

APPENDIX L: T-Chart Case Study

In this appendix, as part of future research I provide two T-chart case studies gathered from student assignments and final design reports. The case studies were selected as representative examples of previously identified behaviors found in biologically inspired design. In the first case study I show how using the analogical evaluation tool, student designers identify several critical incongruities, and respond to the evaluation by significantly altering their problem formulation to align more closely to their analogue. This leads students to change their problem in response to the biological source analogue.

The second case demonstrates a different pattern of behavior, consistent with findings on compound analogical design [3]. In this case, students use multiple biological analogues to solve their design problem. The analogues themselves appear at the outset to range from very similar to entirely dissimilar. Upon close inspection, however, one can see a pattern of sub-problem development, leading to progressively narrower analogies to solve more and more targeted sub-problems. Thus an analogy that appears at the outset to be dissimilar, when framed in the context of the design episode may be interpreted as similar, but only with respect to a small sub-problem within the larger design problem context.

L.1. Polar-paws case study

In the first case study, polar-paws, the design team initially focuses on solving the problem of increasing the grip of automobile tires on snowy and icy surfaces. The team considers a number of systems, but eventually settles on the paws of a polar bear. The students identify an interesting feature of the polar bears paws that they want to use,

described as follows: “Polar bears also have long, stiff hair between the pads of their feet....the hair may allow for the drainage of water from the feet, which will reinforce the adhesion and increase the area of contact with the ice. The hair between the pads absorbs the liquid which is drained off the pads by the animal’s weight while standing. This allows for the hairs to then freeze to the ice allowing for even more friction and traction while running.”

Table L-1. Initial iteration of T-chart comparison between tire-traction problem and polar bear paws.

Design Problem		Polar Bear Paws
Operational Environment		Operational Environment
Ice and snow	Same	Ice and snow
Extreme cold	Same	Arctic Environment
Asphalt	Different	Ice and snow/Cold wet ground
Water	Same	Water
Hilly and Flat Terrain	Same	Hilly and flat terrain
Drivers	N/A	N/A
Functions		Functions
Create Traction	Same	Create traction
Roll on ice and asphalt	Different	Walks with legs
Specifications		Specifications
Non toxic	Same	Non toxic
Light weight	Same	Light weight
Easy to attach and detach	Different	Attached to Polar Bear
Not tear up asphalt	Same	Not leave tracks in snow
Inexpensive	Different	Cost measured in energy used
Fit various sizes of tires	Similar	Size based on bear size
Withstand car weight	Different	Withstand weight of the bear
Performance Criteria		Performance Criteria
Withstand force of at least 3kN	Similar	Withstand weight of polar bear (700-900 kg)
Coefficient of friction at least 0.2	Similar	Polar bear able to move without slipping
Braking distance less than 61.1 m	Similar	Able to stop very quickly

Table L-1 is a reproduction of the initial T-chart generated by the team with respect to their design problem and the target analogy, the polar bear paws. The students provide

the following analysis of transfer challenges (an outcome of the analogical evaluation). *“Performance criteria were difficult to quantify. For one, the tire rolls and the bear walks. The coefficient of friction is calculated in two different ways. Also, tires can slip easier because they are constantly moving. The bear also walks in a very specific way, on the soles of their feet, which increases traction. Tires all move the same way, and very differently from polar bears.”*

In the next design iteration, I detect a significant alteration in the design problem. The designers have moved from the traction problem of automobile tires on snow and ice, to the traction problem of shoes on snow and ice. Table L-2 provides the revised T-chart, in which we see the critical change in the function category, from “roll on ice and asphalt” to “walk in shoes,” with a reevaluation from “different” to “similar”. I infer from the previous paragraph that the change was a direct result of the attention given to the difference in walking versus rolling functions.

A number of other changes are perpetuated throughout the T-chart, for instance, withstand weight of human instead of withstand weight of car, but these changes do not change the similarity assessment. We also note that a performance criteria critical to the car – breaking distance – is no longer considered important to the human-traction problem. This is a possible case of analogical problem evolution; an incremental change in the problem formulation in response to a biological analogue. The problem formulation is simplified (a performance criterion is dropped), and moved incrementally closer to the biological analogue through a function change.

Table L-2. Second iteration of T-chart comparison between shoe-traction problem and polar bear paws

Design Problem		Polar Bear Paws
Operational Environment		Operational Environment
Ice and snow	Same	Ice and snow
Extreme cold	Same	Artic Environment
Asphalt, concrete	Different	Ice and snow/Cold wet ground
Water	Same	Water
Hilly and Flat Terrain	Same	Hilly and flat terrain
Functions		Functions
Create Traction	Same	Create traction
Walk in shoes	Similar	Walks with legs
Specifications		Specifications
Non toxic	Same	Non toxic
Light weight	Same	Light weight (relative to the polar bear)
Easy to attach and detach	Different	Attached to Polar Bear
Not tear up surface	Same	Not leave tracks in snow
Inexpensive	Different	Cost measured in energy used
Fit various sizes and types of shoes	Similar	Size based on bear size
Withstand weight of human	Different	Withstand weight of the bear
Performance Criteria		Performance Criteria
Withstand force of at least 1128.15N	Similar	Withstand weight of polar bear (700-900 kg)
Coefficient of friction at least 0.5	Similar	Polar bear able to move without slipping

L.2. The green light case study

The second case study is focused on the problem of street lights breaking “*because strong materials often are expensive and monolithic materials often have a resonance that can be matched by the wind, causing amplified vibratory motion of the post leading to loosening of joints followed by breakage.*” As in the previous case, the design team identified a number of candidate solutions early on. Three examples seem rather intuitive, the Saguaro cactus, the palm tree and bamboo. Each when evaluated against the problem formulation shows a high degree of similarity.

Table L-3 shows the T-chart for the bamboo, in which we see 8 concepts assessed as

Table L-3. T-chart comparison between the light pole problem and bamboo

Design Problem		Bamboo
Operational Environment		Operational Environment
Outdoors	Same	Outdoors
Wind	Same	Wind
Precipitation	Same	Precipitation
Temperature variation	Similar	Survive in a variety of temperatures
Vibrations from wind and road	Similar	Occasionally deal with these
Functions		Functions
Project light	Similar	Collect light
Elevation of light source	Same	Elevation of light collecting units
Vibration reduction	Similar	Resist breakage
Increase dampening	Similar	Survive high-force winds
Specifications		Specifications
30' Tall	Similar	Up to 30 meters tall
Bright	Different	Green
Requires mast arm	Same	Many small branches extending from node
Decrease materials	Same	Optimize energy expended to support self
Easy installation	Similar	Optimize energy used during growth
Low cost	Same	Optimize total energy use
Performance Criteria		Performance Criteria
Withstand up to 5g's	Different	Unknown
Withstand vortex shedding	Same	Functionally graded
Withstand 70mph winds	Similar	Withstand 120mph winds
Infinite lifespan	Different	Short lived

the same, 8 concepts assessed as similar, and only 3 concepts assessed as different. The Saguaro cactus was even more similar (10 same, 6 similar, 3 different), and the palm tree only slightly less so (6 same, 8 similar, and 5 different). The team used bamboo as an inspiration to create small units from which the light pole could be assembled, and which reduced resonance vibration in the same way as the multi-sectioned bamboo. Likewise they incorporated a mass-damper system similar to palm tree leaves, and incorporated a broad base plate similar to the root system of the Saguaro cactus. I call this incremental addition of biologically inspired features from different sources, compound analogy, and have previously reported that this occurs in roughly half of the projects observed in this

context (Helms, Vattam, Goel, 2009). While the use of compound analogy is not surprising, what we do find surprising are the next two analogues incorporated into the design.

Table L-4. T-chart comparison of the light pole problem and the glass sponge

Design Problem		Glass Sponge
Operational Environment		Operational Environment
Outdoors	Different	Underwater
Wind	Similar	Currents
Precipitation	Similar	Water
Temperature variation	Different	Cold
Vibrations from wind and road	-na-	--
Functions		Functions
Project light	-na-	
Elevation of light source	-na-	
Vibration reduction	Similar	Increases strength
Increase dampening	Similar	Resist breakage
Specifications		Specifications
30' Tall	Different	Up to 2 meters tall
Bright	Different	Transparent
Requires mast arm	Different	No mast arms
Decrease materials	Same	Optimize energy expended to support self
Easy installation	Similar	Optimize energy used during growth
Low cost	Same	Optimize total energy use
Performance Criteria		Performance Criteria
Withstand up to 5g's	Similar	Hierarchical structure to avoid fracturing
Withstand vortex shedding	-na-	
Withstand 70mph winds	-na-	
Infinite lifespan	Same	May live for hundreds to thousands of years

Tables L-4 and L-5 show the T-chart comparisons of the glass sponge, and the peregrine falcon. The glass sponge, as show in Table L-4, has very little in common with the design problem as originally conceptualized which appeared quite similar to the previous, useful analogues. As the students report: “*At first glance, the glass sponge*

doesn't appear to be the best analogy. While it is a cylindrical structure, it faces virtually none of the problems that we were trying to solve. There is no wind in the deep ocean, and very little temperature variation compared to what happens in the air. No source that we found indicates that glass sponges have to deal with resonance at all, though admittedly no source we found looked for resonance either. Even though some sponges have been shown to transmit light, a function that could have been useful in our light pole, they do it by transmitting sunlight down spicules that act like fiber optic strands. Since our light pole is only meant to be on at night, transmitting sunlight would not have worked."

So how and why did the students incorporate the glass sponge, an analogue which they themselves consider highly dissimilar, into their final design? The answer lies understanding the unfolding of the design process. In the following paragraph the students report (emphasis added):

"What makes the glass sponge analogy work is that the multiple layers of hierarchy allow the sponge to create a skeleton that is strong enough to resist damage while remaining lightweight and open enough to conserve materials. We needed a way to strengthen the pole to make up for the aspects of the plants we couldn't copy, so this one aspect of the glass sponge was chosen."

I interpret this statement in the following way. The students began a design with a particular problem in mind. They found obvious analogies to address that problem, and generated a new solution based on that design. After analyzing the new solution, the students realized they had created a new sub-problem – namely that their new solution would be too heavy and consume too much material. This led them to narrow down their

problem formulation and focus on particular specifications, the need to decrease materials and cost, without sacrificing the gains they already made. With this revised focus, the glass sponge, in particular its use of lightweight, low cost materials which, when arranged hierarchically, provided much greater strength and resisted breakage, now appeared as a more similar, and therefore more useful, analogue.

Table L-5. T-chart comparison of the light pole problem and the peregrine falcon.

Design Problem		Peregrine Falcon
Operational Environment		Operational Environment
Outdoors	Same	Outdoors
Wind	Same	Wind
Precipitation	-na-	
Temperature variation	-na-	
Vibrations from wind and road	-na-	
Functions		Functions
Project light	-na-	
Elevation of light source	-na-	
Vibration reduction	Similar	Resist drag during a dive
Increase dampening	-na-	
Specifications		Specifications
30' Tall	Different	Small
Bright	Different	Lightweight
Requires mast arm	Different	Can fly
Decrease materials	-na-	
Easy installation	-na-	
Low cost	-na-	
Performance Criteria		Performance Criteria
Withstand up to 5g's	-na-	
Withstand vortex shedding	-na-	
Withstand 70mph winds	Similar	Reaches 200mph during dive
Infinite lifespan	Different	Short lived

Similarly the peregrine falcon shown in table L-5 was targeted to solve a very small sub-problem in the overall design. The students provide the following rationale:

“A peregrine falcon is very dissimilar to a light pole. It’s not just capable of movement, but the fastest animal on Earth. It can alter its behavior to fit changing conditions, always angling its dive so that it has the smallest profile facing the direction

with the highest wind velocities. If the wind or weather is too much for it, it can always take shelter somewhere until conditions improve. Our light pole cannot adapt to changes into the environment and must withstand them instead. However, this analogy still works because the peregrine falcon is an expert at reducing drag force. While we cannot imitate it completely, it did inspire the profile reduction that led to the very open design of our tower.”

The results from the light pole design problem demonstrate both a strength and weakness of the T-chart analogical evaluation method. On the one hand, as intended the method exposes aspects of the students design process to their own internal scrutiny. In the case of the glass sponge and the peregrine falcon, the students both demonstrate their awareness of the apparent disconnect between their original problem and provide reasonable rationale for their eventual inclusion. On the other hand, these analogues are evaluated against the high-level problem, which exposes the disconnect, but does not allow students to scrutinize the analogues against a more robust sub-problem formulation. In this way, the T-chart (and the four-box method), could be improved by incorporating processes and additional representations to allow for further iterations over sub-problems. Pragmatically, however, this may prove difficult, as time is already at a premium in the context.

These cases studies demonstrate two behaviors which exemplify the richness and complexity of the analogical design processes in biologically inspired design. In the first case, we see that the evaluation of analogies facilitates the evolution of the design problem; drawing the problem closer in similarity to the analogue. In the second case, we see that the evolution of the design problem prompts the establishment of new criteria

against which to judge future analogues. Thus an analogy which may have been previously dismissed or ignored rises in similarity to the smaller sub-problem-at-hand, where it may inspire a sub-solution and often times generate a compound analogy. In an environment in which problems, design solutions, and analogies are constantly interacting, student designers face at least three critical challenges: biological understanding, problem specification, and analogy evaluation.

The tools I developed, based on historical data in the biologically inspired context, provide students with a framework to help navigate these complex dynamics. My experience as documented in this dissertation suggest that the grounded SR.BID and four-box constructs we use in our framework fit naturally into the vocabulary and processes of student designers.

In addition to adoption by students, these case studies provide anecdotal evidence in support of our hypothesis that the T-chart method of analogical evaluation makes salient the differences between the design problem and the biological analogue across dimensions that are important to design. In general I note that students encounter two major pitfalls when transferring concepts from biology to engineering. First there is often difficulty replicating the properties of biological materials. Second, there is often difficulty of anticipating how changes in scale from the biological to the human engineered world will affect the new design. I believe that focusing on differences in specifications (materials, material properties, components, structural connections, etc.) aids students in identifying those areas in which replication of material properties make create design challenges. Additionally, by helping students focus on differences in performance criteria, often differences of one or more orders of magnitude, students are

more quickly away of potential scaling issues. While understanding the implications of these differences requires much deeper comprehension of the mechanisms involved that is provided by our tools, the tools make salient those aspects of the analogy that warrant further investigation.

With respect to the computation of analogical fitness, and in particular with respect to the distance versus feasibility tradeoff, the question now appears less straightforward. The measure of distance depends on the current conceptualization of both the problem and the analogue. Both conceptualizations change over time, often in response to one other. In the two case studies we examined, analogies that were more distant appeared to become less so when the problem was reframed, or when a sub-problem was examined. In the case of the glass sponge and the peregrine falcon, without understanding how those analogues came to be applied, one might judge them as significantly more distant than perhaps they actual were.

Finally, I note that unlisted among our problems is one which has garnered a great detail of attention in biologically inspired design: the problem of finding an appropriate biological analogue. Search technologies in biologically inspired design tend to use single word and/or single concept searches, often focused on function-related keywords or design strategies. Some technologies also provide translations of functional keywords between domains. If the goal of search is to find the best analogy for a given design problem, and if the retrieved analogue is to be evaluated across multiple dimensions of the problem as our framework suggests, then I believe the next generation search technologies will have to take these multiple dimensions into account.

REFERENCES

- Adelson, B., & Solway, E. (1988) A Model of Software Design, in Chi, M, R Glaser and M Farr (eds.), *The Nature of Expertise*, Erlbaum, Hillsdale, NJ, USA.
- Altshuller, G. S. (1984). *Creativity as an exact science: The theory of the solution of inventive problems* (Vol. 5). CRC Press.
- Atman, C., & Chimka, J. (1999) A Comparison of Freshman and Senior Engineering Design Processes. *Design Studies* 20(2):131-152.
- Ausubel, D. P. (2000). *The acquisition and retention of knowledge: A cognitive view*. Springer.
- Ball, L., Evans, J., Dennis, I., Ormerod, T., (1997) Problem-Solving Strategies and Expertise in *Engineering Design. Thinking and Reasoning* 3(4):247-270.
- Bar-Cohen, Y. (Editor, 2006) *Biomimetics: Biologically Inspired Technologies*. Taylor & Francis.
- Bar-Cohen, Y. (Editor, 2011) *Biomimetics: Nature-Based Innovation*. Taylor & Francis.
- Bartoli, G., Borsani, A., Borri, C., Martelli, A., Procino, L., & Vezzosi, A. (2009) Leonardo, the wind and the flying sphere. *EACWE*, 5, 1.
- Benyus, J. (1997) *Biomimicry: Innovation Inspired by Nature*. New York: William Morrow.
- Bhatta, S., Goel, A., & Prabhakar, S. (1994). Innovation in analogical design: A model-based approach. In *Artificial Intelligence in Design'94* (pp. 57-74). Springer Netherlands.
- Bhatta, S., & Goel, A. (1996) Model-Based Indexing and Index Learning in Engineering Design. *Engineering Applications of AI, Special issue on Machine Learning in Engineering*, 9(6):601-610.
- Bhatta, S. R., & Goel, A. K. (1997). A functional theory of design patterns. In *IJCAI (1)* (pp. 294-300).
- Bonser, R. H. (2006). Patented biologically-inspired technological innovations: a twenty year view. *Journal of Bionic Engineering*, 3(1), 39-41.
- Bonser, R., & Vincent, J. (2007) Technology Trajectories, Innovation, and the Growth of Biomimetics. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, pp. 1177-1180
- Bransford, J. D., Brown, A. L., & Cocking, R. R. (2000). Learning and transfer. *How*

- people learn: Brain, mind, experience, and school*, 51-78.
- Bybee, R. W. (1997). *Achieving scientific literacy: From purposes to practices*. Heinemann, 88 Post Road West, PO Box 5007, Westport, CT 06881
- Chakrabarti, A., Sarkar, P., Leelavathamma, B., & Nataraju, B. (2005) A Functional Representation for Aiding Biomimetic and Artificial Inspiration of New Ideas. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 19:113-132.
- Chandrasekaran, B., (1990) Design Problem Solving: A Task Analysis, *AI Magazine*, 11(4), pp.59-71.
- Chandrasekaran, B., Goel, A. K., & Iwasaki, Y. (1993). Functional representation as design rationale. *Computer*, 26(1), 48-56.
- Chandrasekaran, B. (1994). Functional representation: A brief historical perspective. *Applied Artificial Intelligence an International Journal*, 8(2), 173-197
- Chandrasekaran, B., & Josephson, J. R. (2000). Function in device representation. *Engineering with computers*, 16(3-4), 162-177.
- Cheong, H., Chiu, I., Shu, L., Stone, R., McAdams, D. (2011) Biologically Meaningful Keywords for Functional Terms of the Functional Basis. *Journal of Mechanical Design*, 133.
- Chiu, I., and Shu, L. H., (2007) Biomimetic Design Through Natural Language Analysis to Facilitate Cross-domain Information Retrieval. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 21:45-59.
- Christiaans, H (1992) *Creativity in design* PhD Thesis, Delft University of Technology, Delft, The Netherlands.
- Christensen, B. T., & Schunn, C. D. (2007). The relationship of analogical distance to analogical function and preinventive structure: The case of engineering design. *Memory & cognition*, 35(1), 29-38.
- Clement, J. J. (Ed.). (2008). Creative model construction in scientists and students: the role of imagery, analogy, and mental stimulation. Springer.
- Cross, N. (2001) Design Cognition: Results of Protocol and Other Empirical Studies of Design Activity. In C. M. Eastman, W. M. McCracken, & W. C. Newstetter (Eds.), *Design Knowing and Learning: Cognition in Design Education*. (pp. 79-103). Amsterdam: Elsevier.
- Cross, N. (2004) Expertise in Design: an Overview. *Design Studies* 25(5):427-441.
- Cross, N & Clayborn Cross, A. (1998) Expertise in Engineering Design, *Research in*

- Engineering Design* 10:141-149.
- Dahl, O.J., Dykstra, E.W., & Hoare, C.A.R. (1972). *Structured programming*. New York: Academic Press.
- Darden, L., & Cook, M. (1994, January). Reasoning strategies in molecular biology: abstractions, scans and anomalies. In *PSA: Proceedings of the Biennial Meeting of the Philosophy of Science Association* (pp. 179-191). Philosophy of Science Association.
- Darke, J. (1979) The Primary Generator and the Design Process. *Design Studies*, 1(1):36-44.
- Dinar, M., Shah, J., Hunt, G., Campana, E., & Langley, P. (2011). Towards a formal representation model of problem formulation in design. In *Proceedings of ASME DETC*.
- Dinar, M., Maclellan, C., Danielescu, A., Shah, J., & Langley, P. (2012). Beyond Function-Behavior-Structure. In *Proceedings of the Fifth International Conference on Design Computing and Cognition*.
- Dörner, D. (1980). On the difficulties people have in dealing with complexity. *Simulation & Gaming*, 11(1), 87-106.
- Dorst, K. (2003) The Problem of Design Problems. In N. Cross & E. Edmonds (Eds.), *Expertise in Design* (pp. 135-147). Sydney: Creativity and Cognition Studio Press.
- Dorst, K., & Cross, N. (2001) Creativity in the Design Process: Co-evolution of Problem-Solution. *Design Studies*, 22(5):425-437.
- Dunbar, K. (1995). How scientists really reason: Scientific reasoning in real-world laboratories. *The nature of insight*, 396, 73.
- Dunbar, K. (2001). The analogical paradox: Why analogy is so easy in naturalistic settings yet so difficult in the psychological laboratory. *The analogical mind: Perspectives from cognitive science*, 313-334.
- Dym, C. L., & Brown, D. C. (2012). *Engineering design: Representation and reasoning*. Cambridge University Press.
- Encyclopedia of Life (2013). Available from <http://www.eol.org>. Accessed Jan 15, 2013.
- Biomimicry 3.8 Institute (2013). Available from <http://ben.biomimicry.net/curricula-and-resources/university-curricula/> Accessed Jan 15, 2013.
- Erden, M. S., Komoto, H., van Beek, T. J., D'Amelio, V., Echavarria, E., & Tomiyama, T. (2008). A review of function modeling: Approaches and applications. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 22(02), 147-169

- Falkenhainer, B., Forbus, K.D., Gentner, D. (1989) The structure-mapping engine: Algorithm and examples. *Artificial Intelligence*, 41(1): 1-63.
- Feltovich, P., Coulson, R., Spiro, R., & Dawson-Saunders, B. K. (1992). Knowledge Application and Transfer for Complex Tasks in 111-Structured Domains: Implications for Instruction and Testing in Biomedicine. *Advanced models of cognition for medical training and practice*, 97, 213.
- French, M. (1998). *Conceptual design for engineers*. Springer.
- Funke, J. (2001). Dynamic systems as tools for analysing human judgement. *Thinking & Reasoning*, 7(1), 69-89.
- Gentner, D. (1983) Structure-mapping: A Theoretical Framework for Analogy. *Cognitive Science*, 7(2): 155-170.
- Gellert, E. (1962). Children's conceptions of the content and functions of the human body. *Genetic Psychology Monographs*.
- Gero, J.S. (1993). Towards a Model of Exploration in Computer-Aided Design, In J.S. Gero and N. Tyugu (Eds.) *Formal Design Methods for Computer-Aided Design*, North-Holland, Amsterdam.
- Gero, J. S. (1990). Design Prototypes: A Knowledge Representation Schema for Design. *AI magazine*, 11(4), 26.
- Gero, J. S. (1996). Creativity, emergence and evolution in design. *Knowledge-Based Systems*, 9(7), 435-448.
- Gero, J. S. (2002). Computational models of creative designing based on situated cognition. In *Proceedings of the 4th conference on Creativity & cognition* (pp. 3-10). ACM.
- Gero, J. S., & Kannengiesser, U. (2004). The situated function–behaviour–structure framework. *Design studies*, 25(4), 373-391.
- Gick, M. L., & Holyoak, K. J. (1983). Schema induction and analogical transfer. *Cognitive psychology*, 15(1), 1-38.
- Glaser, B. G., & Strauss, A. L. (2009). *The discovery of grounded theory: Strategies for qualitative research*. Transaction Books.
- Goel, A. K. (1992). Integrating case-based and model-based reasoning: a computational model of design problem solving. *AI Magazine*, 13(2), 50.
- Goel, A. K. (1997). Design, Analogy and Creativity. *IEEE Expert*, 12(3): 62-70.
- Goel, A. K., & Bhatta, S. R. (2004). Use of design patterns in analogy-based design.

- Advanced Engineering Informatics*, 18(2), 85-94.
- Goel, A. K., & Chandrasekaran, B. (1989). Functional Representation of Designs and Redesign Problem Solving. In *IJCAI* (pp. 1388-1394).
- Goel, A. K., & Chandrasekaran, B. (1992) Case-Based Design: A Task Analysis. In *Artificial Intelligence Approaches to Engineering Design, Volume II: Innovative Design*, C. Tong and D. Sriram (editors), pp. 165-184, San Diego: Academic Press, 1992.
- Goel, A. K., de Silva Garza, A. G., Grué, N., Murdock, J. W., & Recker, M. M. (1997). Functional explanations in design. *Program of the IJCAI'97 Workshop on Modeling and Reasoning about Function*, pp 221-234.
- Goel, A. K., Rugaber, S., & Vattam, S. (2009). Structure, behavior, and function of complex systems: the structure, behavior, and function modeling language. *AI EDAM*, 23(1), 23-35.
- Goel, A. K., McAdams, D. A., & Stone, R. B. (2013). *Biologically Inspired Design: Computational Methods and Tools*.
- Goel, V., & Pirolli, P. (1992) The Structure of Design Problem Spaces. *Cognitive Science*, 16: 395-429.
- Gross, M. D., & Do, E. L. (1995, October). Diagram query and image retrieval in design. In *Image Processing, 1995. Proceedings., International Conference on* (Vol. 2, pp. 308-311). IEEE.
- Guindon, R. (1990) Designing the Design Process: Exploiting Opportunistic Thoughts. *Human Computer Interaction*, 5:305-344.
- Helms, M., Vattam, S., Goel, A. (2008) Compound Analogical Design, or How to Make a Surfboard Disappear. In B.C. Love, K. McRae, & V.M. Sloutsky (Eds.) *Proceedings of the 30th Annual Conference of the Cognitive Science Society*, Washington D.C.:Cognitive Science Society
- Helms, M., Vattam, S., & Goel, A. (2009) Biologically Inspired Design: Products and Processes. *Design Studies*, 30(5):606-622.
- Helms M, Vattam S, Goel A (2010) The Effects of Functional Modeling on Understanding Complex Biological Systems. *Proc. ASME Int. Design Engineering Technical Conf.*, Montreal, Canada.
- Helms, M., Vattam, S. Goel, A., Yen, J., Weissburg, M. (2008) Problem-Driven and Solution-Based Design: TwinProcesses of Biologically Inspired Design. In *Silicon + Skin: Biological Processes and Computation, Proceedings of the 28th Annual Conference of the Association for Computer Aided Design in Architecture (ACADIA)*, Minneapolis, MN.

- Hillier, W., Musgrove, J., & O'Sullivan, P., (1972) Knowledge and Design, in Mitchell, W J (Ed.) *Environmental Design Research and Practice*, University of California, US.
- Hmelo, C. E., Holton, D. L., & Kolodner, J. L. (2000). Designing to learn about complex systems. *The Journal of the Learning Sciences*, 9(3), 247-298.
- Ho, C-H., (2001) Some Phenomena of Problem Decomposition Strategy for Design Thinking:Differences Between Novices and Experts, *Design Studies*, 22(1): 27-45.
- Hofstadter, D. R. (1996). Analogy-making, fluid concepts, and brain mechanisms. *Connectionism, Concepts, and Folk Psychology: The Legacy of Alan Turing. Vol. II*, 195-247.
- Holyoak, K. J. (1991) Symbolic Connectionism: Toward Third-generation Theories of Expertise, in Ericsson, K A and J. Smith (eds.) *Toward a General Theory of Expertise: Prospects and Limits*, Cambridge University Press, Cambridge, UK.
- Holyoak, K.J., Thagard, P. (1989) Analogical Mapping by Constraint Satisfaction. *Cognitive Science* 13(3): 295-355.
- Hundal, M. S. (1990). A systematic method for developing function structures, solutions and concept variants. *Mechanism and Machine theory*, 25(3), 243-256.
- Keane, M. T. (1988). *Incremental Analogising: Theory and model*. Human Cognition Research Laboratory.
- Kirschman, C. F., Fadel, G. M., & Jara-Almonte, C. (1998). Classifying functions for mechanical design. *Transactions-ASME Journal of Mechanical Design*, 120, 475-482.
- Kolodner, J. L. (Ed.). (1993). *Case based learning* (Vol. 10, No. 3). Springer.
- Krugar, C., Cross, N., (2006) Solution Driven Versus Problem Driven Design: Strategies and Outcomes. *Design Studies* 27(5): 527-548.
- Kurz-Milcke, E., Nersessian, N. J., & Newstetter, W. C. (2004). What has history to do with cognition? Interactive methods for studying research laboratories. *Journal of Cognition and Culture*, 4(3-4), 3-4.
- Lamp, J. & Milton, S. (2007) Grounded Theory As Foundations For Methods In Applied Ontology, *Proceedings of the 4th International Conference on Qualitative Research in IT & IT in Qualitative Research (QualIT)*, Victoria University of Wellington.
- Lave, J., & Wenger, E. (1991). *Situated learning: Legitimate peripheral participation*. Cambridge university press.
- Lawson, B. (1979) Cognitive Strategies in Architectural Design *Ergonomics* 22(1): 59-

68.

- Liikkanen, L. & Perttula, M. (2008) Exploring Problem Decomposition in Conceptual Design Among Novice Designers. *Design Studies* 30(1): 38-59.
- Linsey, J.S., Wood, K.L., and Markman, A.B. (2008) Modality and representation in analogy. *Artificial Intelligence for Design, Analysis and Manufacturing* 22(02): 85-100.
- Lloyd, P., & Scott, P. (1994) Discovering the design problem, *Design Studies*, 15 (2): 125-140.
- Maher, M. L., Poon, J., & Boulanger, S. (1996) Formalising Design Exploration as Co-evolution: a Combined Gene Approach, in J S Gero and FSudweeks (eds) *Advances in formal design methods for CAD*, Chapman and Hall, London, UK.
- Mak, T.W. & Shu, L.H. (2004). Abstraction of Biological Analogies for Design. *CIRP Annals – Manufacturing Technology* 53(1):117 – 120.
- Maslow, A. H. (1966). The psychology of science: A renaissance. *Gateway Editions, Ltd, South Bend, IN*, 50.
- Miles, M. B., & Huberman, A. M. (1994). *Qualitative data analysis: An expanded sourcebook*. Sage.
- Mintzes, J. J., Trowbridge, J. E., Arnaudin, M. W., & Wandersee, J. H. (1991). Children's biology: Studies on conceptual development in the life sciences. *The psychology of learning science*, 179-202.
- Nagel, R., Midha, P, Tinsley, A, Stone, R., McAdams, D., Shu, L. (2008) Exploring the Use of Functional Models in Biomimetic Concept Design. *ASME Journal of Mechanical Design*, 130(2).
- Nagel, J., Stone, R., McAdams, D. (2010a) Exploring the Use of Category and Scale to Scope a Biological Functional Model. *Proc. ASME Int. Design Engineering Technical Conf.*, Montreal, Canada.
- Nagel, J., Stone, R., McAdams, D. (2010b) An Engineering-to-Biology Thesaurus for Engineering Design. *Proc. ASME Int. Design Engineering Technical Conf.*, Montreal, Canada.
- Newell, A., & Simon, H. A. (1972). *Human problem solving* (Vol. 14). Englewood Cliffs, NJ: Prentice-Hall.
- Nersessian, N.J., & Chandrasakaran, S. (2009) Hybrid Analogies in Conceptual Innovation in Science. *Cognitive Systems Research*, 10(3): 178-188.
- Nersessian, N. (2008). *Creating scientific concepts*. The MIT Press.

- Pahl, G., & Beitz, W. (2003) *Engineering design* (K. M. Wallace, Trans.). London: The Design Council.
- Qian, L., & Gero, J. S. (1996). Function-behavior-structure paths and their role in analogy-based design. *AI EDAM*, 10(4), 289-312.
- Reitman, W. R. (1964) Heuristic Decision Procedures, Open Constraints and the Structure of Ill-defined Problems. In M.W.Shelly & G.L. Bryan (Eds.) *Human Judgments and Optimality*, New York, Wiley.
- Restrepo, J and Christiaans, H (2004) Problem Structuring and Information Access. *Design Journal of Design Research* (4).
- Rittel, H., Webber, M. (1974) Dilemmas in a General Theory of Planning, *Policy Sciences*, 4: 155-169.
- Roozenburg, N.F.M. and Eekels, J. (1995) *Product Design: Fundamentals and Methods*. Wiley, Chichester.
- Rowe, P. G. (1987) *Design thinking*. Cambridge, MA: The MIT Press.
- Sarkar, P. & Chakrabarti, A. (2008) The Effect of Representation of Triggers on Design Outcomes. *Artificial Intelligence for Design, Analysis and Manufacturing*, 22(02): 101-116.
- Schön, D A (1983) *The Reflective Practitioner*, Temple-Smith, London, UK.
- Schön, D A (1988) Designing: Rules, Types and Worlds, *Design Studies*, 9: 181-190.
- Shu, L., Stone, R., McAdams, D, Greer, J. (2007) Integrating Function-Based and Biomimetic Design for Automatic Concept Generation. In *Proc. International Conference on Engineering Design (ICED'07)*, Paris, France, August 2007.
- Shu, L. H., Ueda, K., Chiu, I., & Cheong, H. (2011). Biologically inspired design. *CIRP Annals-Manufacturing Technology*, 60(2), 673-693.
- Simon, H. A. (1973) The Structure of Ill-structured Problems. *Artificial Intelligence*, 4:181-201.
- Singh, L. H., Hansen, H. N., Gegeckaitė, A., Moon, J., and Chan, C. (2006) Case Study in Biomimetic Design: Handling and Assembly of Microparts, *Proceedings of the ASME IDETC/CIE*, Philadelphia, PA.
- Strauss, A. L., & Corbin, J. (1990). *Basics of qualitative research* (Vol. 15). Newbury Park, CA: Sage publications.
- Stone, R. B., & Wood, K. L. (2000). Development of a functional basis for design. *Journal of Mechanical Design*, 122, 359.

- Sturges, R. H., O'Shaughnessy, K., & Kilani, M. I. (1996). Computational model for conceptual design based on extended function logic. *Artificial Intelligence for Engineering, Design, Analysis and Manufacturing*, 10(04), 255-274.
- Umeda, Y., & Tomiyama, T. (1997). Functional reasoning in design. *IEEE expert*, 12(2), 42-48.
- Umeda, Y., Ishii, M., Yoshioka, M., Shimomura, Y., & Tomiyama, T. (1996). Supporting conceptual design based on the function-behavior-state modeler. *AIEDAM*, 10(4), 275-288.
- Vattam, S., Helms, M., and Goel, K. (2007) Biologically-Inspired Innovation in Engineering Design: A Cognitive Study. Technical Report, Graphics, Visualization and Usability Center, Georgia Institute of Technology GIT-GVU-07-07.
- Vattam, S., Helms, M., & Goel, A. (2009) Nature of Creative Analogies in Biologically Inspired Innovative Design. In *Proc. Seventh ACM Conference on Creativity & Cognition*, Berkeley, California
- Vattam,S., Wiltgen,B., Helms,M., Goel, A. & Yen,J. (2010) DANE: Fostering Creativity in and through Biologically Inspired Design. In *Proc. First International Conference on Design Creativity*, Kobe, Japan.
- Vattam, S., & Goel, A. (2011) Foraging for inspiration: Understanding and supporting the information seeking practices of biologically inspired designers To appear in *Proc. ASME IDETC Conference on Design Theory and Methods*, Washington DC.
- Vincent, J. F., & Mann, D. L. (2002). Systematic technology transfer from biology to engineering. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 360(1791), 159-173
- Vincent, J.F.V., Bogatyreva, O.A., Bogatyrev, N.R., Bowyer, A., and Pahl, A.-K. (2006) Biomimetics: Its Practice and Theory, *Journal of the Royal Society Interface*,3:471-482.
- Vogel, S. (2000). *Cats' paws and catapults: Mechanical worlds of nature and people*. WW Norton & Company
- Vygotsky, L. (1978). Interaction between learning and development. *Readings on the development of children*, 34-41.
- Wilson, J.O. (2008) A Systematic Approach to Bio-inspired Conceptual Design. PhD Dissertation. Georgia Institute of Technology, Atlanta, USA.
- Wilson, J.O., and Rosen, D. (2007) Systematic Reverse Engineering of Biological Systems. *Proc. 2007 ASME IDETC/CIE*, Las Vegas, Nevada.
- Wirth, N. (1971) Program Development by Stepwise Refinement. *Communications of the*

ACM,14(4): 221-227.

- Wood-Robinson, C. (1995). Children's biological ideas: Knowledge about ecology, inheritance, and evolution. *Learning science in the schools: Research informing practice*, 111-131.
- Yaner, P. W., & Goel, A. K. (2002). Retrieving 2-D line drawings by example. In *Diagrammatic Representation and Inference* (pp. 97-99). Springer Berlin Heidelberg.
- Yen, J., & Weissburg, M. (2007) Perspectives on Biologically Inspired Design: Introduction to the Collected Contributions. *Journal of Bioinspiration and Biomimetics*, 2.
- Yen, J., Helms, M., Vattam, S., & Goel, A. K. (2010). Evaluating biological systems for their potential in engineering design. *Advances in Natural Science*, 3(2), 27-40.
- Yen, J., Weissburg, M., Helms, M., & Goel, A. (2011). Biologically inspired design: a tool for interdisciplinary education. *Biomimetics: Nature-Based Innovation*.
- Yen, J., Helms, M., Goel, A., Tovey, C., Weissburg, M. (2013) Adaptive Evolution of Teaching Practices in Biologically Inspired Design. In A. Goel, D. McAdams & R. Stone (editors), *Biologically Inspired Design*, pp. 153-199, London: Springer-Verlag.