TASK SCHEDULING IN SUPERCAPACITOR BASED ENVIRONMENTALLY POWERED WIRELESS SENSOR NODES

A Dissertation
Presented to
The Academic Faculty

by

Hengzhao Yang

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology
August 2013

TASK SCHEDULING IN SUPERCAPACITOR BASED ENVIRONMENTALLY POWERED WIRELESS SENSOR NODES

Approved by:

Dr. Ying Zhang, Advisor
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Ghassan AlRegib
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Mary Ann Weitnauer
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Gabriel Alfonso Rincón-Mora
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Yang Wang
School of Civil and Environmental Engineering
Georgia Institute of Technology

Date Approved: May 2, 2013

To my family,

for their support and encouragement.

ACKNOWLEDGEMENTS

I would like to express my deepest appreciation to my advisor, Dr. Ying Zhang, for her support and guidance throughout my Ph.D. program.

I owe my sincere gratitude to the other members of my dissertation defense committee: Dr. Ghassan AlRegib, Dr. Mary Ann Weitnauer, Dr. Gabriel Alfonso Rincón-Mora, and Dr. Yang Wang, for their services and comments on my research.

I am very grateful to all the professors whose classes I have taken, especially to those who have generously allowed me to audit their classes: Dr. Ian F. Akyildiz, Dr. Farrokh Ayazi, Dr. Maysam Ghovanloo, Dr. Vincent J. Mooney, Dr. Arijit Raychowdhury, and Dr. Hua Wang, for broadening my horizons.

I am thankful to all the colleagues and friends in Savannah and in Atlanta for enriching my life.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

The objective of this dissertation is to develop task scheduling guidelines and algorithms for wireless sensor nodes that harvest energy from ambient environment and use supercapacitor based storage systems to buffer the harvested energy. This dissertation makes five contributions. First, a physics based equivalent circuit model for supercapacitors is developed. The variable leakage resistance (VLR) model takes into account three mechanisms of supercapacitors: voltage dependency of capacitance, charge redistribution, and self-discharge. Second, the effects of time and supercapacitor initial state on supercapacitor voltage change and energy loss during charge redistribution are investigated. Third, the task scheduling problem in supercapacitor based environmentally powered wireless sensor nodes is studied qualitatively. The impacts of supercapacitor state and energy harvesting on task scheduling are examined. Task scheduling rules are developed. Fourth, the task scheduling problem in supercapacitor based environmentally powered wireless sensor nodes is studied quantitatively. The modified earliest deadline first (MEDF) algorithm is developed to schedule nonpreemptable tasks without precedence constraints. Finally, the modified first in first out (MFIFO) algorithm is proposed to schedule nonpreemptable tasks with precedence constraints. The MEDF and MFIFO algorithms take into account energy constraints of tasks in addition to timing constraints. The MEDF and MFIFO algorithms improve the energy performance and maintain the timing performance of the earliest deadline first (EDF) and first in first out (FIFO) algorithms, respectively.

CHAPTER I

INTRODUCTION

This dissertation considers the task scheduling problem in wireless sensor nodes that harvest energy from ambient environment and buffer the scavenged energy using supercapacitor based storage systems. This chapter introduces the background of this dissertation. Section 1.1 defines the terms in the dissertation title. The term "environmentally powered wireless sensor nodes" refers to wireless sensor nodes powered by energy harvested from environment. The energy storage systems of these wireless sensor nodes use "supercapacitor based" configurations. The "task scheduling" problem in supercapacitor based environmentally powered wireless sensor nodes is studied. Section 1.2 summarizes the objective of this dissertation and outlines the work presented in this dissertation.

Starting with the Distributed Sensor Networks program at the Defense Advanced Research Projects Agency (DARPA) around 1980 [13], sensor networks have been evolving into a key technology. Driven by the advances in microelectromechanical systems (MEMS), wireless communications, and digital electronics [4], inexpensive, multifunctional, and low power sensor nodes such as Mica [24], PicoRadios [51], and Smart Dust [73] have been developed to perform sensing, data processing, and communicating tasks. Wireless sensor networks (WSNs) have been deployed for various applications [4] including military surveillance [23], habitat monitoring [68], biomedical health monitoring [40], underground WSNs [3], underwater WSNs [2], and multimedia WSNs [1].

1.1 Background

1.1.1 Power Subsystems of Wireless Sensor Nodes

A wireless sensor node is an integrated system composed of multiple components: micro-controller, communication subsystem, sensor/actuator subsystem, storage subsystem, and power subsystem [25]. In this dissertation, the power subsystem is considered and the other subsystems are taken as load. While the size and cost of electronics are scaling down, the power subsystem is usually the largest and most expensive part of a sensor node [58]. Meanwhile, the power subsystem is also the lifetime limiting factor of a sensor node [58]. Depending on the energy source characteristics, the power subsystems can be classified into three groups: energy reservoirs, power distribution, and energy harvesting [25, 58].

A sensor node can be powered by the energy stored in energy reservoirs. Non-rechargeable primary batteries such as zinc air, lithium, and alkaline chemistries are examples of such energy reservoirs. This power subsystem is the most predominant means of providing energy to sensor nodes. The benefits of this way include that non-rechargeable batteries are relatively inexpensive and that sensor nodes can be located anywhere without requiring the existing power infrastructure. However, using non-rechargeable batteries can be problematic in that the lifetime of sensor nodes is limited due to the limited battery capacity. For example, the lifetime of the Great Duck Island sensor node is limited to about two months [68].

The second way to supply energy to sensor nodes is to distribute power to the nodes from a nearby energy rich source. Sensor nodes powered by energy distributed in wired or wireless manners have been proposed. A wired back channel is originally used to reprogram or monitor a sensor node. As a side effect, the sensor node can be powered through the wire. Depending on the energy source, two categories of wire-powered sensor nodes are the Power-over-Ethernet (PoE)-powered nodes and USB-powered nodes. PoE-powered nodes such as Mirage [14] and MicaZ-based MoteLab

[74] are powered through the wire if the programming board can take power from the PoE cable. USB-powered nodes including Omega test bed [50] and TWIST [22] are powered through the USB host. Although it is easy to power a sensor node using a back wire, this power solution only works for deployments where a PoE or USB wiring is available. The wireless way to distribute power is to use radio frequency (RF) radiation. In [21], electronic identification tags are powered by a nearby energy rich source that transmits RF energy to the tags. This approach is less efficient in densely deployed sensor networks where a large area must be flooded with RF radiation.

Wireless sensor nodes can also be powered by energy harvested from ambient environment. A sensor node powered by renewable energy is expected to run for a long period of time. A renewable energy powered sensor node usually consists of energy source, energy collector, energy storage, and regulator. The renewable energy is converted into electric energy by the energy collector from the energy source. The energy can directly power the load or be stored in the energy storage. A regulator may be used to match the operating ranges of different components. The amount of harvested energy depends on how long the energy source is in operation and the energy source power density. Various energy harvesting technologies have been developed. For example, solar energy [30], mechanical vibration [42], RF radiation [34], human power [60], thermoelectric energy [66], and wind energy [69]. Among these energy sources, the outdoor solar energy is popular for two reasons: it has high power density and solar panels are commercially available.

In this dissertation, the third category of power subsystems is considered. Wireless sensor nodes powered by energy harvested from ambient environment are called "environmentally powered wireless sensor nodes".

1.1.2 Energy Buffers: Rechargeable Batteries and Supercapacitors

For environmentally powered wireless sensor nodes, the energy harvested from ambient environment is the primary power source. Usually, secondary energy buffers such as rechargeable batteries and supercapacitors are needed to store the harvested energy. The energy buffer characteristics [25,30,31] and sensor nodes [67] using these buffers to store the harvested energy are reviewed in this section.

The NiMH battery is one of the most popular energy buffers for wireless sensor network applications because it has relatively high energy density, simple charging method, and low cost. Wireless sensor nodes using NiMH batteries include Heliomote [54], Fleck [16], HydroWatch [70], and the node in [5].

The Li-ion battery has the highest energy density and high charge-discharge efficiency. While these characteristics make this chemistry a good candidate for wireless sensor network applications, the complicated charging mechanism is a limiting factor. Systems using the Li-ion battery need either a dedicated charging management chip or a software package to correctly control the battery. The Li-ion chemistry is used in sensor nodes such as ZebraNet [77].

The NiCd battery is similar to the NiMH battery in that both chemistries have similar charging and discharging characteristics, and that both chemistries are available in standard form factors such as AA. The NiCd chemistry is inferior to the NiMH chemistry due to its smaller capacity and the memory effect that causes its capacity to decrease over multiple uses. Therefore, the NiMH battery is preferred over the NiCd battery for wireless sensor network applications.

The lead acid battery is the most commonly used energy storage in high power systems due to its large capacity and simple charging control. However, this battery is not usually used for low power wireless sensor network applications because of its small energy density.

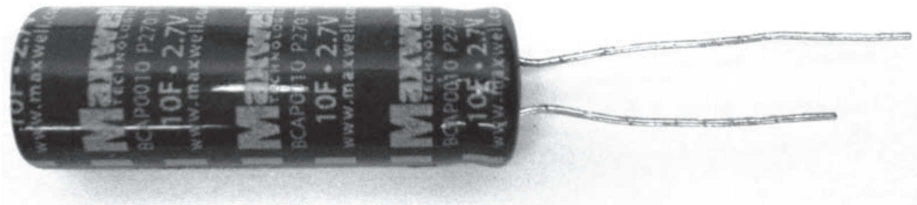


Figure 1: A 10 F supercapacitor.

A supercapacitor (also known as ultracapacitor or electric double-layer capacitor (EDLC)) is a capacitor with capacity high enough to serve as energy buffer for wireless sensor network applications. Figure 1 shows a 10 F supercapacitor manufactured by Maxwell. Its model number is BCAP0010. The nominal capacitance and voltage of this supercapacitor is 10 F and 2.7 V, respectively. This supercapacitor is used as a sample in this dissertation. A supercapacitor has a long cycle life, high charge-discharge efficiency, and fast charge-discharge characteristic. However, the supercapacitor leakage rate is also high. Sensor nodes such as Everlast [62], TwinStar [79], and the solar harvester in [8] use supercapacitors as energy buffer.

In general, a rechargeable battery has high capacity and low leakage rate. However, its relatively short cycle life limits the lifetime of a sensor node. The cycle life of a rechargeable battery is defined as the number of charge-discharge cycles before its capacity falls below 80% of its rated capacity [31]. For instance, the cycle life of the NiMH and Li-ion chemistries is 500-800 cycles and 1000-1200 cycles, respectively [31]. On the other hand, a supercapacitor has a much longer cycle life, which can be more than 500,000 cycles [62]. To leverage the complementary strengths of rechargeable batteries and supercapacitors, sensor nodes using hybrid energy storage systems such as Prometheus [26] (Li-polymer and supercapacitor), Trio [18] (Li-ion and supercapacitor), and the node in [48] (Li-ion and supercapacitor) have been developed.

This dissertation considers environmentally powered wireless sensor nodes that use supercapacitor based energy storage systems, which are called "supercapacitor based environmentally powered wireless sensor nodes".

1.1.3 Power Management in Battery Based Sensor Nodes

Energy efficiency is a major concern in wireless sensor networks [53]. Considerable efforts have been made to develop power management strategies for various applications [57,75,78]. Depending on the characteristics of the power subsystems of wireless sensor networks, power management goals can be different. For sensor nodes powered by energy reservoirs such as primary batteries, the power management goals are usually to minimize energy consumption [43, 64] or to maximize expected lifetime [35, 59, 63, 76] while certain performance requirements are met.

Environmentally powered wireless sensor networks may have different power management goals and policies. The harvested energy may be used in two modes. First, the harvested energy is treated as a supplement to the energy stored in rechargeable batteries or in supercapacitors. Therefore, the power management objective is still to maximize system lifetime. Second, the harvested energy is used by a sensor node at an appropriate rate such that the node can operate perpetually, which is called energy neutral operation in [27]. Sensor nodes can achieve energy neutral operation while an expected performance level is supported (subject to hardware failure). For environmentally powered sensor nodes, the power management goals are to achieve energy neutral operation and to maximize performance level [27].

Most existing works on power management in environmentally powered wireless sensor nodes focus on rechargeable battery based energy storage systems. As shown in [28,72], energy harvesting aware decisions improve the system performance compared to battery aware decisions for the specific applications considered. The concept of energy neutral operation of a solar energy powered sensor node is proposed in [27]. A heuristic algorithm is proposed to decrease the sensor node duty cycle when the harvested energy is low and to increase the duty cycle when the harvested energy is high. The work in [27] is improved by [71]. An energy management framework for energy harvesting embedded systems is developed in [45]. A set of algorithms is developed for various application scenarios: real time scheduling, application rate control, and reward maximization. Dynamic voltage and frequency scaling (DVFS) is introduced to further improve the system energy efficiency in [38].

Power management problems for supercapacitor based and hybrid energy storage systems are not well investigated. A switching policy between supercapacitor and rechargeable battery is developed in [26]. In [79], a supercapacitor leakage aware energy synchronization framework is presented for a supercapacitor based energy storage system. Sizing and topology reconfiguration strategies are optimized for charging and discharging multiple supercapacitors in [29].

In this dissertation, the task scheduling problem in supercapacitor based environmentally powered wireless sensor nodes is considered. A task refers to an event in a wireless sensor node that draws certain amount of energy from the energy storage system. Task scheduling is a fundamental power management problem because the typical operations in a wireless sensor node such as sensing the environment, processing the collected data, and transmitting the data packets can all be interpreted as different tasks.

1.2 Dissertation Objective and Outline

The objective of this dissertation is to develop task scheduling guidelines and algorithms for wireless sensor nodes that harvest energy from ambient environment and use supercapacitor based storage systems to buffer the harvested energy. This dissertation makes five contributions. First, a physics based equivalent circuit model for supercapacitors is developed. The variable leakage resistance (VLR) model takes into account three mechanisms of supercapacitors: voltage dependency of capacitance, charge redistribution, and self-discharge. Second, the effects of time and supercapacitor initial state on supercapacitor voltage change and energy loss during charge redistribution are investigated. Third, the task scheduling problem in supercapacitor based environmentally powered wireless sensor nodes is studied qualitatively. The impacts of supercapacitor state and energy harvesting on task scheduling are examined. Task scheduling rules are developed. Fourth, the task scheduling problem in supercapacitor based environmentally powered wireless sensor nodes is studied quantitatively. The modified earliest deadline first (MEDF) algorithm is developed to schedule nonpreemptable tasks without precedence constraints. Finally, the modified first in first out (MFIFO) algorithm is proposed to schedule nonpreemptable tasks with precedence constraints. The MEDF and MFIFO algorithms take into account energy constraints of tasks in addition to timing constraints. The MEDF and MFIFO algorithms improve the energy performance and maintain the timing performance of the earliest deadline first (EDF) and first in first out (FIFO) algorithms, respectively.

The remaining part of this dissertation is organized as follows. Chapter II presents the VLR model for supercapacitors. Chapter III analyzes supercapacitor charge redistribution and energy loss. Chapter IV studies the effects of supercapacitor state and energy harvesting on task scheduling. Chapter V proposes the MEDF algorithm for nonpreemptable tasks without precedence constraints. Chapter VI proposes the MFIFO algorithm for nonpreemptable tasks with precedence constraints. Chapter VII concludes this dissertation.

CHAPTER II

SUPERCAPACITOR MODELING AND CHARACTERIZATION

This chapter presents the variable leakage resistance (VLR) model for supercapacitors. Section 2.1 reviews supercapacitor physics and three supercapacitor models based on which the VLR model is developed: energy iteration equation (EIE) model, two branch model, and three branch model. Section 2.2 analyzes supercapacitor self-discharge and presents the VLR model. The VLR model is an equivalent circuit model that takes into account three mechanisms of supercapacitors: voltage dependency of capacitance, charge redistribution, and self-discharge. Section 2.3 demonstrates the characterization procedures for the VLR model parameters. Section 2.4 evaluates the four supercapacitor models (VLR, EIE, two branch model, and three branch model) by comparing the simulated voltages using these models and the measured voltage during three experiments: charging-redistribution, self-discharge, and dynamic charging-discharging experiments. Section 2.5 is a summary.

2.1 Related Work

2.1.1 Supercapacitor Physics

A supercapacitor is constructed with two porous activated carbon electrodes impregnated with electrolyte and separated by a porous insulating membrane [10], as shown in Figure 2. When a voltage is applied to the supercapacitor terminals, a double layer is formed at the interface between the electrode and the electrolyte [52]. The energy storage mechanism is primarily electrostatic rather than Faradaic [52]. It is possible that a pseudocapacitive component also contributes to the total capacitance [10]. One

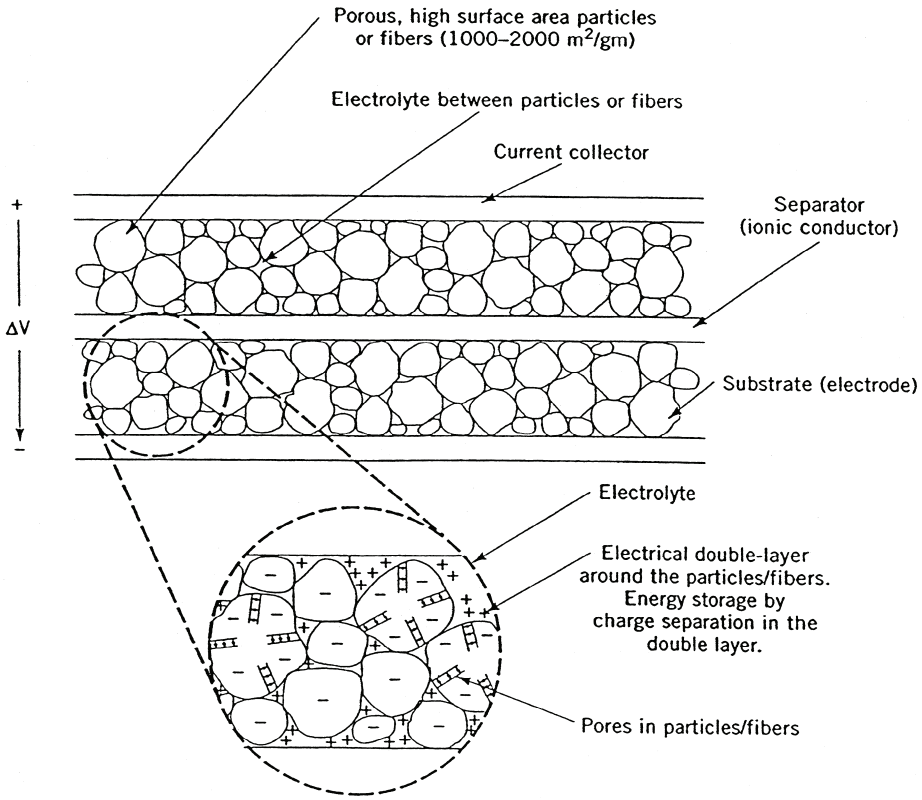

Porous, high surface area particles or fibers (1000–2000 m²/gm)
Electrolyte between particles or fibers
Current collector
Separator (ionic conductor)
+
ΔV
−
Substrate (electrode)
Electrolyte
Electrical double-layer around the particles/fibers. Energy storage by charge separation in the double layer.
Pores in particles/fibers

Figure 2: Supercapacitor structure [10].

of the most important internal characteristics of a supercapacitor is the porous nature of its electrodes [49]. The internal kinetic effects of the supercapacitor electrodes are different from those observed in the conventional planar electrodes due to the high porosity of the supercapacitor electrodes [19]. Consequently, the supercapacitor impedance cannot be modeled by a simple RC circuit. Instead, a supercapacitor can be modeled by a distributed parameter system [32]. The interface electrochemistry suggests that a complex RC network can describe the behavior of supercapacitors [32].

Supercapacitors can be characterized using the electrochemical impedance spectroscopy (EIS) technique, which is a general approach to characterize energy storage devices by measuring their complex impedances [9]. The nature of impedances in various frequency ranges can be determined by analyzing the frequency dependencies of the real part and the imaginary part [33]. Various equivalent circuit models [7, 11, 19, 49, 52, 56] have been developed. They use the porous electrode theory to interpret the impedance spectrum of a supercapacitor. With the assumption of homogeneous electrode pore size, a general impedance model consists of three impedances linking to electrode, electrolyte and electrode/electrolyte interface, respectively [7]. The general impedance model can be modified if interface roughness [61] or pore size randomness [65] is considered. The general impedance model can be approximated by a set of N interleaved RC branches. However, it is usually very difficult to determine more than five or six independent parameters efficiently considering their strong influences on each other [9].

Alternatively, supercapacitors can be characterized in time domain by conducting various experiments such as constant power tests and constant current tests [33]. This approach is often used to develop an equivalent circuit model [6,17,20,80] to describe the terminal behavior of a supercapacitor. These models usually contain at least two RC branches to take into account the species diffusion phenomena (also called the long range phenomena) during supercapacitor relaxation [7]. In these models, one RC branch models the fast dynamics and the other branches model the slower dynamics. Therefore, each RC branch has a different time constant. During supercapacitor relaxation, the charge stored in a supercapacitor tends to redistribute among different RC branches to achieve a balanced state and the supercapacitor terminal voltage changes. In this dissertation, charge redistribution refers to the species diffusion phenomena during supercapacitor relaxation. Supercapacitor charge redistribution is originated from the distributive nature of supercapacitor impedance and the species diffusion phenomena during supercapacitor relaxation. The supercapacitor activated carbon electrodes are composed of pores of randomly distributed sizes [65]. Pores of different sizes have different impedances. Therefore, after a charging or discharging action, the charge stored in a supercapacitor redistributes among different pores to reach a balanced state.

2.1.2 Supercapacitor Models

2.1.2.1 Energy Iteration Equation (EIE) Model

The supercapacitor model used in wireless sensor node energy storage system design in [79] is called the energy iteration equation (EIE) model in this dissertation. The EIE model is developed based on the leakage power profile of a supercapacitor. After a supercapacitor is fully charged, its terminal voltage is measured during a self-discharge experiment. As given by Eq. (1), the remaining energy E stored in a supercapacitor can be calculated from its terminal voltage V and rated capacitance C:

$$E = \frac{1}{2}CV^2. \tag{1}$$

Taking into account the harvested energy and consumed energy, the energy stored in the supercapacitor at the beginning of every T-second time slot is calculated as follows:

$$E(n+1) = E(n) + E_H(n) - E_C(n) - P(n) * T, \tag{2}$$

where $E(n+1)$ and $E(n)$ are the remaining energy at the beginning of the $(n+1)$-th and n-th time slots, $E_H(n)$ and $E_C(n)$ are the harvested energy and consumed energy during the n-th time slot, $P(n)$ is the leakage power corresponding to $E(n)$ and is treated as a constant during a time slot, respectively. The supercapacitor voltage $V(n+1)$ at the beginning of the $(n+1)$-th time slot can be calculated from the remaining energy as follows:

$$V(n+1) = \sqrt{\frac{2E(n+1)}{C}}. \tag{3}$$

Supercapacitor leakage power profile can be approximated using a piecewise linear function Eq. (4):

$$P = \begin{cases} c_1 * E + d_1, & E_1 \leq E < E_2, \\ c_2 * E + d_2, & E_2 \leq E < E_3, \\ c_3 * E + d_3, & E_3 \leq E \leq E_4. \end{cases} \tag{4}$$

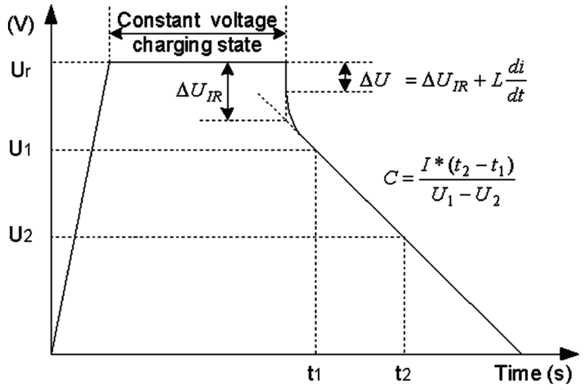

(V)
Constant voltage charging state
Ur
U1
U2
ΔU_{IR}
$\Delta U = \Delta U_{IR} + L\frac{di}{dt}$
$C = \frac{I*(t_2 - t_1)}{U_1 - U_2}$
t1
t2
Time (s)

Figure 3: Supercapacitor voltage during constant current charging or discharging experiments [12].

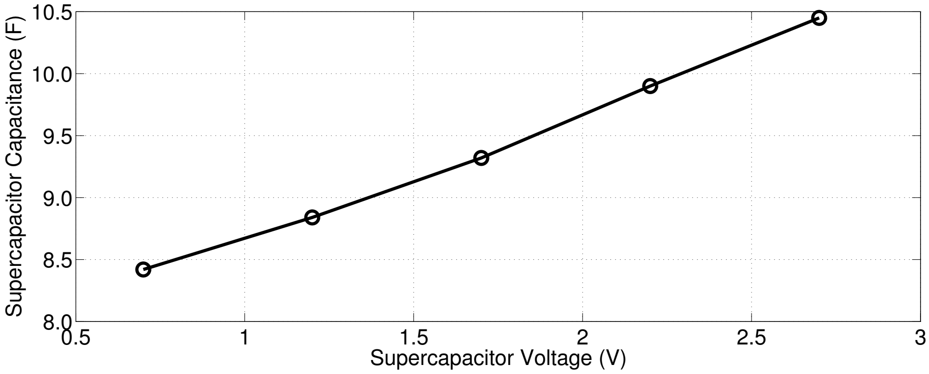


Figure 4: Supercapacitor capacitance versus voltage.

The EIE model has two problems. First, the EIE model assumes that supercapacitor capacitance is always equal to its nominal capacitance. In fact, supercapacitor capacitance depends on its terminal voltage. Supercapacitor nominal capacitance is characterized using Eq. (5) according to the standard IEC 62391, as shown in Figure 3. Figure 4 shows the capacitances of the 10 F sample supercapacitor (Figure 1) measured at various terminal voltages. This voltage dependency of capacitance can be approximately characterized using a linear function.

$$C = \frac{I * (t_2 - t_1)}{U_1 - U_2} \tag{5}$$

The second problem is that the EIE model assumes that supercapacitor voltage drop is only due to self-discharge when no external charging or discharging current is present. A supercapacitor can be modeled as a network of multiple *RC* branches.

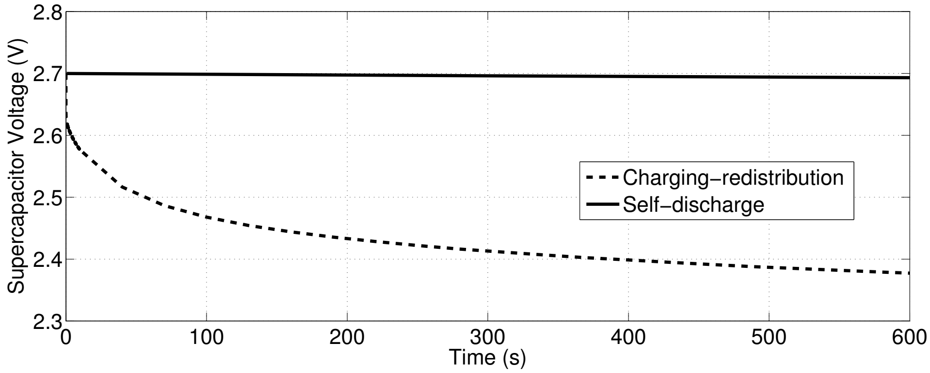


Figure 5: Supercapacitor voltage drops during charging-redistribution and self-discharge experiments.

The time constants of the RC branches are different. After a charging or discharging action, the voltages across the capacitances are usually different. Therefore, charge redistribution among different branches takes place. The supercapacitor terminal voltage changes during charge redistribution. Figure 5 compares the measured supercapacitor voltages during a self-discharge experiment and a charging-redistribution experiment. For the self-discharge experiment, the supercapacitor voltage is measured after the supercapacitor is charged using a 2.7 V voltage source for 1 h. For the charging-redistribution experiment, the fully discharged supercapacitor is charged using a 1 A current source to 2.7 V. After the charging current is disconnected, the supercapacitor voltage is measured during the following charge redistribution phase. Figure 5 shows that charge redistribution results in a much larger voltage drop than self-discharge.

2.1.2.2 Two Branch Model

The two branch model shown in Figure 6 is an equivalent circuit model [20]. The first branch is composed of a resistance R_1 and a differential capacitance C_1. The differential capacitance C_1 includes a constant capacitance C_0 and a voltage dependent capacitance $K_V * V$. The total capacitance is $C_1 = C_0 + K_V * V$. The first branch dominates the immediate behavior of a supercapacitor in response to a charging or

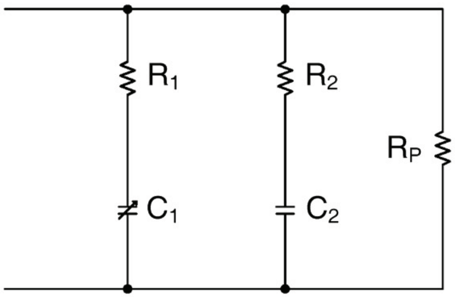


Figure 6: Supercapacitor two branch model [20].

discharging action. This branch captures the voltage dependency of capacitance. The second branch including R_2 and C_2 models charge redistribution. The equivalent parallel resistance R_P represents the effect of self-discharge.

2.1.2.3 Three Branch Model

The three branch model shown in Figure 7 is proposed to better model supercapacitor self-discharge [17]. In the two branch model, supercapacitor self-discharge is represented by the equivalent parallel resistance R_P and only the internal ohmic leakage is considered. In the three branch model, the third branch composed of R_r and C_r is introduced to represent the self-discharge due to diffusion controlled Faradaic redox reactions. The leakage current due to diffusion controlled Faradaic redox reactions is proportional to the concentration gradient of the diffusible redox species [17]. The time dependence of the concentration gradient at a particular distance from a plane electrode is generally inversely proportional to the square root of time, which leads to a decreasing self-discharge rate.

2.2 Supercapacitor Modeling

2.2.1 Supercapacitor Self-discharge

A charged supercapacitor is in a state of high Gibbs energy [47]. Therefore, a thermodynamic "driving force" results in spontaneous decline of Gibbs energy [47]. This

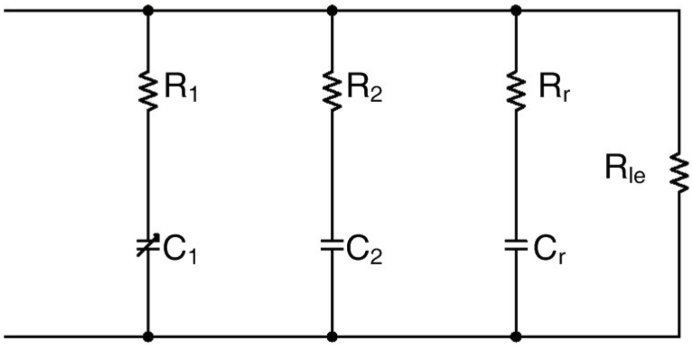


Figure 7: Supercapacitor three branch model [17].

Figure 8: Maccor supercapacitor testing system.

decline manifested as decay in supercapacitor voltage is called self-discharge. Self-discharge rate, which is usually diminishing with time, determines the shelf life of a supercapacitor [15]. Figure 8 shows the Maccor supercapacitor testing system. The measured supercapacitor terminal voltage of the 10 F sample supercapacitor is shown in Figure 9. The supercapacitor is first charged to its rated voltage 2.7 V using a constant voltage source for 1 h. The supercapacitor terminal voltage is measured for the following 12 h after the voltage source is disconnected.

Open circuit self-discharge of a supercapacitor must take place through coupled anodic and cathodic processes to pass parasitic currents at one or both electrodes since

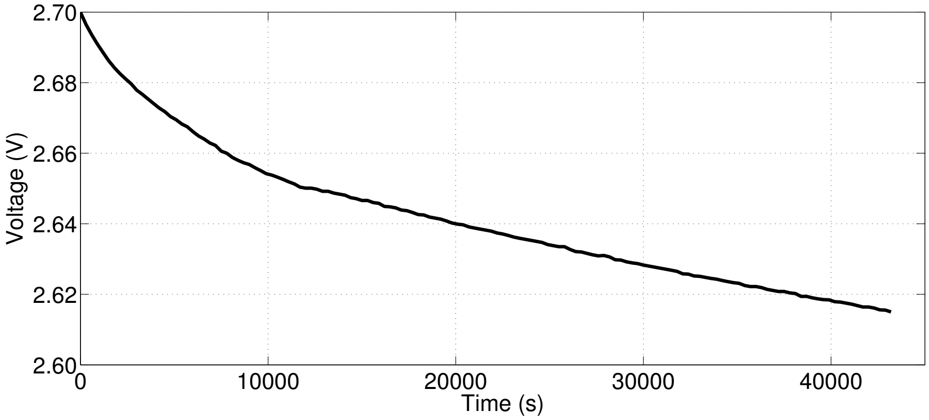


Figure 9: Measured supercapacitor voltage during a self-discharge experiment.

there is no external circuit through which discharge can pass [47]. Supercapacitor self-discharge can be ascribed to three mechanisms [15, 39, 47, 55]:

1. A Faradaic charge transfer reaction can occur if the voltage across an electrolyte-carbon interface exceeds the decomposition potential limit of the electrolyte. This action results in a self-discharge process that has a voltage dependent Faradaic resistance.

2. A diffusion controlled Faradaic process involving depolarization by impurity reactions, which appears to dominate self-discharge in the first few hours. Accumulation of an excess ionic concentration can occur near the electrolyte-carbon interface if a supercapacitor is charged to a threshold voltage. When the supercapacitor is disconnected from the charging source, part of the charge undergoes self-discharge because of the presence of impurities. A simplified relationship between the supercapacitor voltage during self-discharge and its initial voltage is represented by the following equation:

$$V = V_0 - m\sqrt{t}, \tag{6}$$

   where V is the supercapacitor voltage during self-discharge, V_0 is the initial voltage, m is the diffusion parameter, and t is the self-discharge time.

3. A leakage current can arise through the double layer at the electrolyte-carbon interface if a supercapacitor has internal ohmic leakage pathways. The characteristic behavior of this self-discharge mechanism is modeled as follows:

$$V = V_0 e^{-t/R_L C}, \tag{7}$$

where R_L is a constant leakage resistance and C is the supercapactitor capacitance.

Under normal operation conditions, the first self-discharge mechanism does not need to be modeled because supercapacitor voltage is below the electrolyte decomposition potential limit. The second mechanism dominates self-discharge in the first few hours. Then the internal ohmic leakage becomes the dominant factor, which causes supercapacitor voltage to decay exponentially [55]. As shown in Figure 10, a single exponential function (the dashed line) does not match the 12-h measurement. However, after 7 h, the supercapacitor voltage can be fitted well using an exponential function with a fixed time constant (the dash-dotted line). When this function is extended to the first 7 h, the fitting function deviates significantly from the measurement. Based on this observation, a variable leakage resistance can be adopted to model self-discharge for the first few hours. Afterwards, the leakage resistance can be assumed to be a constant.

2.2.2 Variable Leakage Resistance (VLR) Model

As shown in the previous sections, the EIE model takes into account the time varying supercapacitor self-discharge, the two branch model and three branch model take into account supercapacitor voltage dependency of capacitance and charge redistribution. The proposed variable leakage resistance (VLR) model shown in Figure 11 combines the respective benefits of two modeling approaches. The VLR model is a simplified equivalent circuit model. The first branch has three components: resistance R_1, a constant capacitance C_0, and a voltage dependent capacitance $K_V * V$. The total

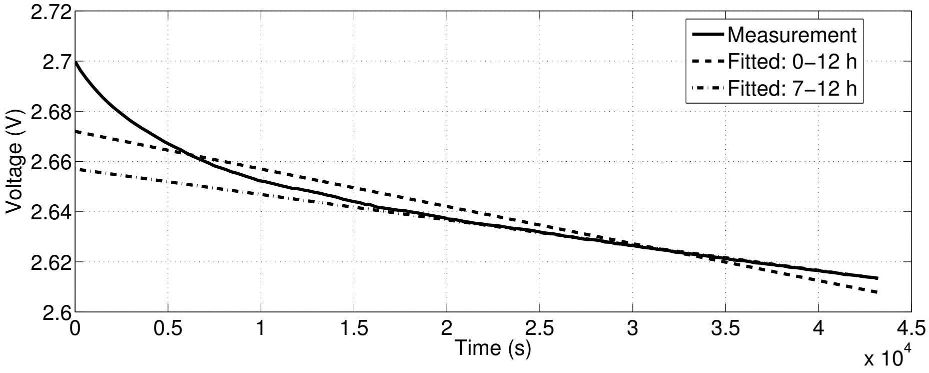


Figure 10: Fitting supercapacitor voltage during self-discharge experiment using exponential functions.

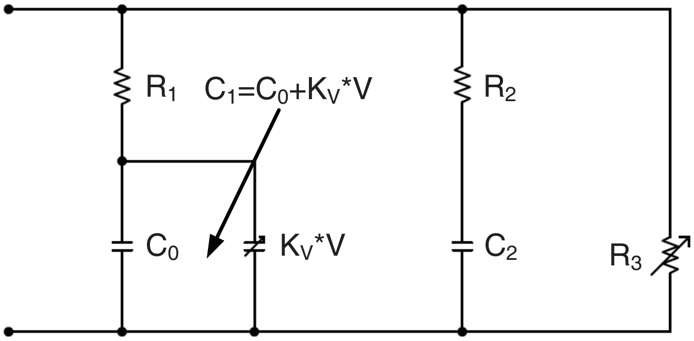


Figure 11: Supercapacitor variable leakage resistance (VLR) model.

capacitance of the first branch is $C_1 = C_0 + K_V * V$. This branch models voltage dependency of capacitance. Its time constant is in the order of seconds. The second branch includes R_2 and C_2. This branch models charge redistribution with a time constant of minutes. The variable leakage resistance R_3 models the time varying self-discharge.

2.3 Supercapacitor Characterization

This section demonstrates the procedures to characterize the VLR model parameters. The VLR model parameters can be determined by performing two experiments: a charging-redistribution experiment for the first and second branches, and a self-discharge experiment for R_3. The 10 F Maxwell supercapacitor is used as an example in this dissertation to illustrate the characterization procedures.

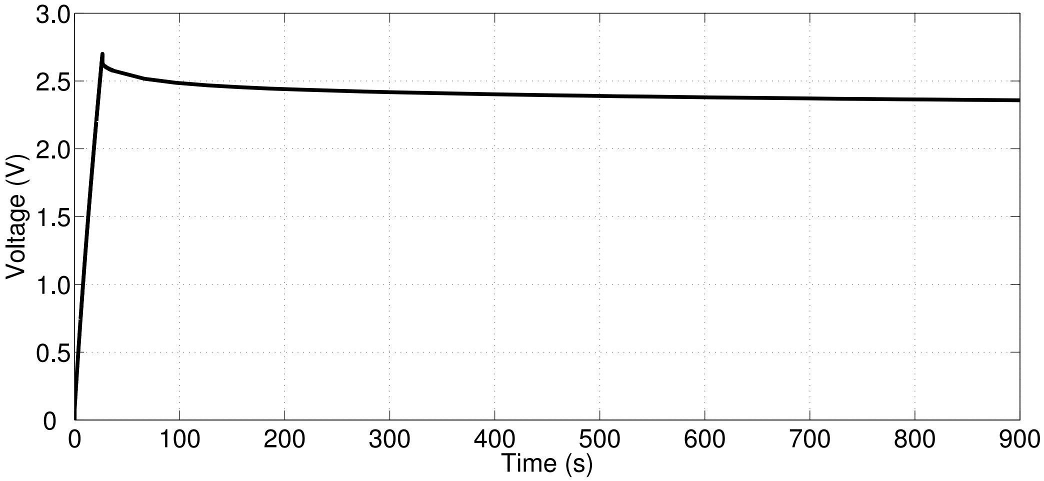


Figure 12: Measured supercapacitor voltage during a charging-redistribution experiment. Charging current is 1 A.

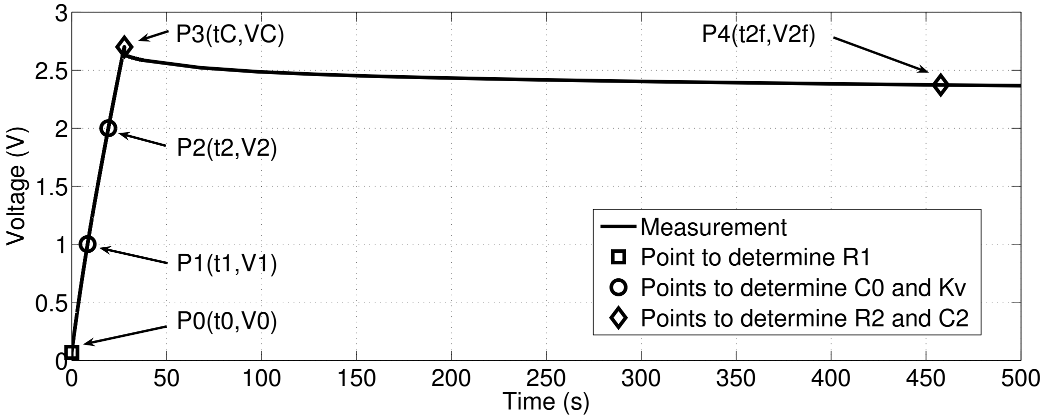


Figure 13: Characterization of the first and second branches of VLR model.

2.3.1 Characterization of First and Second Branches

The VLR model parameters of the first and second branches are determined using the approach in [20]. Figure 12 shows the measured supercapacitor terminal voltage during a charging-redistribution experiment. The fully depleted supercapacitor is charged to 2.7 V using a 1 A constant current source. The charging current is then disconnected and the supercapacitor experiences charge redistribution. The total experiment time is 900 s. The data points used to determine the model parameters are marked in Figure 13.

2.3.1.1 R_0

The resistance R_0 is calculated using data point $P_0(t_0, V_0)$. In Eq. (8), V_0 is the voltage difference between the supercapacitor terminals at the beginning of the charging-redistribution experiment. The charging current is denoted by I_C.

$$R_1 = \frac{V_0}{I_C} \tag{8}$$

2.3.1.2 C_0 *and* K_V

The constant capacitance C_0 and linear capacitance coefficient K_V are determined using data points $P_1(t_1, V_1)$ and $P_2(t_2, V_2)$. Assuming that all the charge is injected into the first branch during the charging process, the current-voltage relationship for C_1 is given by Eq. (9):

$$i = \frac{dq}{dt} = \frac{dq}{dv}\frac{dv}{dt} = (C_0 + K_V v)\frac{dv}{dt}. \tag{9}$$

In case of a constant charging current I_C, the time-voltage relationship for the supercapacitor can be derived from Eq. (9):

$$t = f(V) = \frac{C_0}{I_C}V + \frac{K_V}{2I_C}V^2. \tag{10}$$

Given two data points $P_1(t_1, V_1)$ and $P_2(t_2, V_2)$ in the measured supercapacitor voltage profile during the charging phase, the following equations must hold:

$$\begin{cases} t_1 = \dfrac{C_0}{I_C}V_1 + \dfrac{K_V}{2I_C}{V_1}^2, \\ t_2 = \dfrac{C_0}{I_C}V_2 + \dfrac{K_V}{2I_C}{V_2}^2. \end{cases} \tag{11}$$

The values of C_0 and K_V can be solved from Eq. (11):

$$\begin{cases} C_0 = \left(\dfrac{t_1}{V_1} - \dfrac{V_1 t_2 - t_1 V_2}{V_2^2 - V_1 V_2}\right) I_C, \\ K_V = 2\left(\dfrac{V_1 t_2 - t_1 V_2}{V_1 V_2^2 - V_1^2 V_2}\right) I_C. \end{cases} \tag{12}$$

Table 1: VLR Model Parameter Values of a 10 F Supercapacitor

$R_1(\Omega)$	C_0(F)	K_V(F/V)	$R_2(\Omega)$	C_2(F)
0.0677	7.011	1.042	64.52	1.825

2.3.1.3 C_2 and R_2

The second branch parameters C_2 and R_2 are determined using data points $P_3(t_C, V_C)$ and $P_4(t_{2f}, V_{2f})$. After the supercapacitor reaches its rated voltage, the charging current is removed. Part of the charge stored in the first branch is transferred to the second branch during the charge redistribution phase. It is assumed that the voltages across the capacitances C_1 and C_2 are same at time $t = 3\tau_2$, where $\tau_2 = R_2C_2$ is the second branch time constant. The value of τ_2 is determined by observing the supercapacitor voltage during the charge redistribution phase. After a certain period of time, the supercapacitor voltage changes very slowly and it can be assumed that charge redistribution is completed. This period of time is assumed to be $t = 3\tau_2$. In this experiment, $t = 3\tau_2$ is approximately 400 s. The data points $P_3(t_C, V_C)$ and $P_4(t_{2f}, V_{2f})$ are the ending points of the charging and charge redistribution phases, respectively. Assuming that the charge injected into the supercapacitor is conserved during the charging and redistribution phases, C_2 can be determined as follows:

$$Q = I_C T_C = C_2 V_{2f} + (C_0 + \frac{K_V}{2} V_{2f}) V_{2f}, \tag{13}$$

where T_C is the charging time. Finally, R_2 is calculated using Eq. (14):

$$R_2 = \frac{\tau_2}{C_2}. \tag{14}$$

The VLR model first and second branch parameter values are listed in Table 1.

2.3.2 Supercapacitor Self-discharge Analysis

Supercapacitor terminal voltage decreases during self-discharge. Figure 14 shows the circuit used to analyze supercapacitor terminal voltage change during self-discharge.

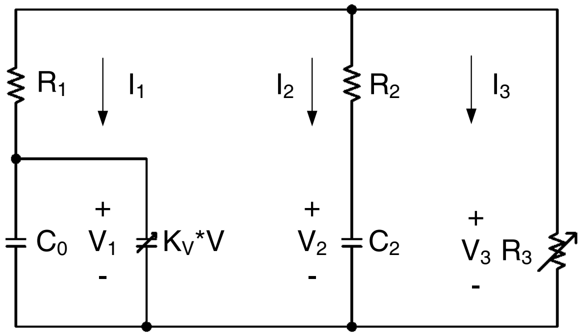


Figure 14: Circuit for self-discharge analysis.

The three branch currents are related by the Kirchhoff's Current Law:

$$I_3 = -I_1 - I_2. \tag{15}$$

Once the current I_3 is determined, together with the measured supercapacitor terminal voltage V_3 during self-discharge, the resistance R_3 can be calculated using the Ohm's Law:

$$R_3 = \frac{V_3}{I_3}. \tag{16}$$

The first branch current I_1 is also the current through the capacitance C_1:

$$I_1 = \frac{d(C_1V_1)}{dt} = \frac{d((C_0 + K_VV_1)V_1)}{dt} = (C_0 + 2K_VV_1)\frac{dV_1}{dt}. \tag{17}$$

The voltage V_1 across the capacitance C_1 is related to V_3 as follows:

$$V_1 = V_3 - I_1R_1. \tag{18}$$

Similarly, the current and voltage relationships for the second branch are:

$$I_2 = \frac{d(C_2V_2)}{dt} = C_2\frac{dV_2}{dt}, \tag{19}$$

$$V_2 = V_3 - I_2R_2. \tag{20}$$

Since the circuit element voltages and currents are tangled, it is difficult to derive the analytical solutions for I_3 and R_3. A numerical approach is adopted to solve the system composed of Eqs. (15)-(20).

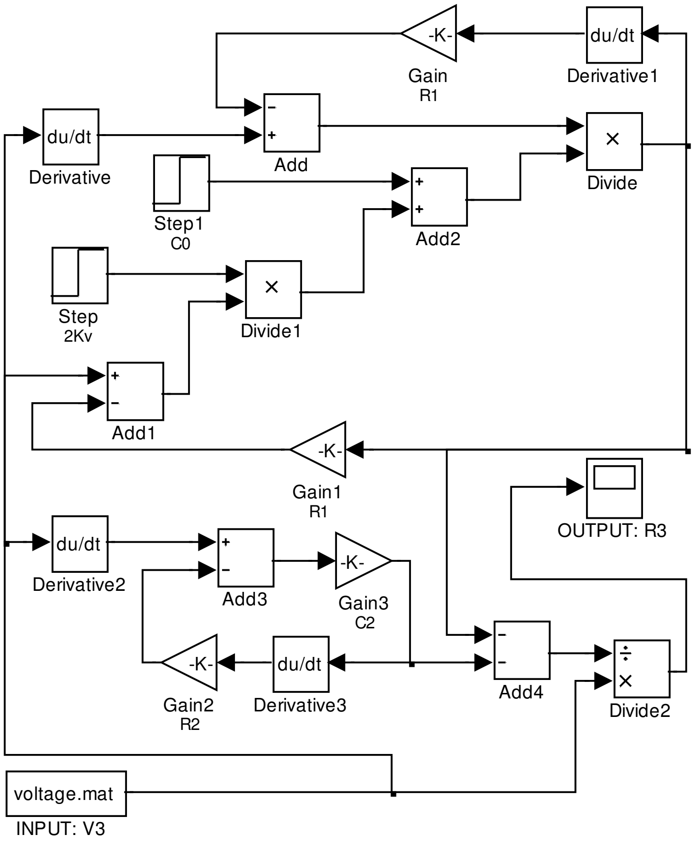


Figure 15: MATLAB Simulink model for R_3 calculation.

2.3.3 Supercapacitor Self-discharge Characterization

As shown in Figure 15, a MATLAB Simulink model is implemented to determine R_3 from the system composed of Eqs. (15)-(20). The input is the supercapacitor terminal voltage V_3 measured during self-discharge. The output is the variable leakage resistance R_3. The blocks representing the VLR model parameters are annotated. For example, the constant capacitance C_0 is represented by the block “Step1”.

The variable leakage resistance R_3 is determined by feeding the supercapacitor terminal voltage (Figure 9) measured in the 12-h self-discharge experiment to the Simulink model. As shown in Figure 16, the VLR value varies with self-discharge time. The VLR value increases from 11,000 Ω to 175,920 Ω in the first 7 h (0-25,920 s), and stays at this value till the end of this experiment (at 43,200 s). The increase in VLR value during the first 7 h is the result of combined self-discharge mechanisms,

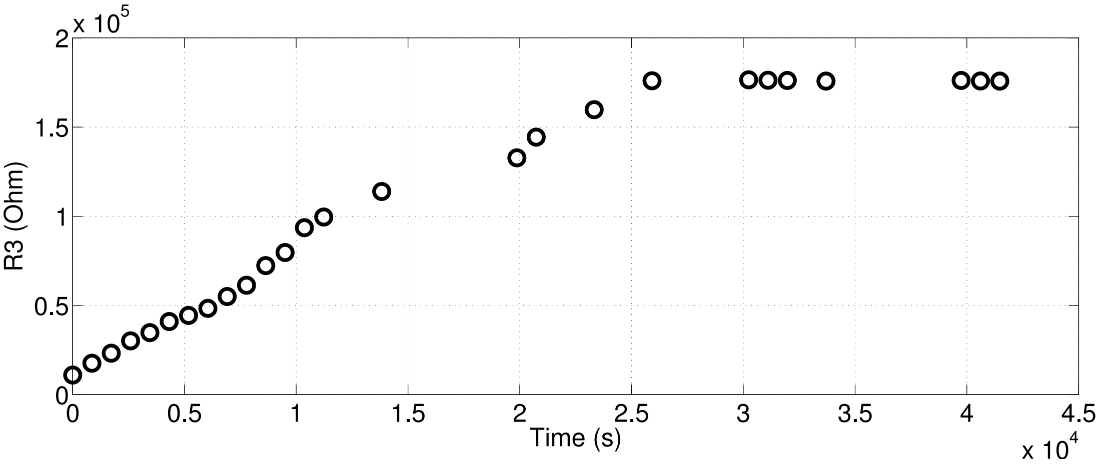


Figure 16: Calculated R_3 versus self-discharge time.

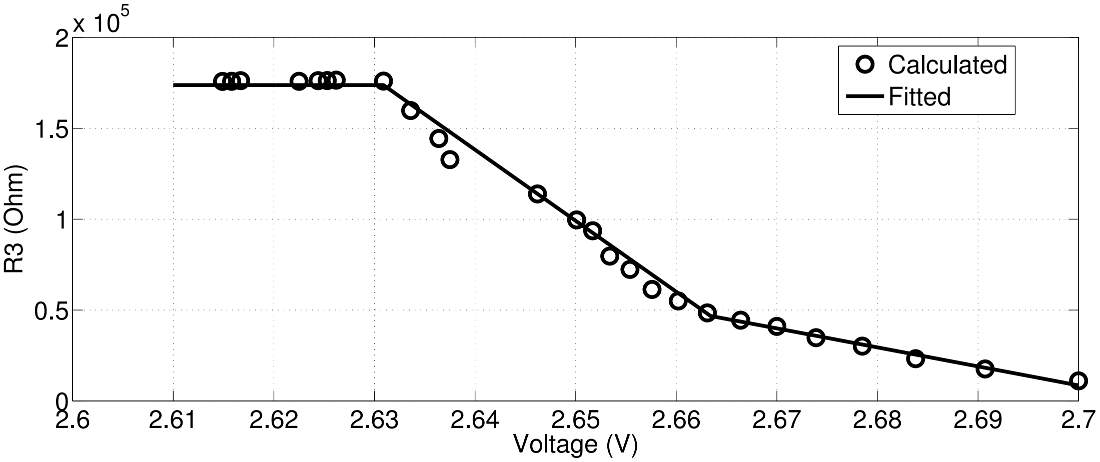


Figure 17: Calculated and fitted R_3 versus supercapacitor voltage.

which include diffusion controlled Faradaic process and internal ohmic leakage. The diffusion controlled Faradaic process dominates. After 7 h, the internal ohmic leakage takes over and a constant leakage resistance accounts for the exponential decay of supercapacitor voltage.

Since supercapacitor voltage decreases when self-discharge time increases, the VLR value can also be related to supercapacitor voltage, as shown in Figure 17. The VLR value increases from 11,000 Ω to 175,920 Ω when supercapacitor voltage decreases from 2.7 V to 2.6309 V. This voltage range corresponds to the first 7 h (0-25,920 s) of the self-discharge experiment.

Although the VLR value can be expressed as a function of either self-discharge time or supercapacitor voltage, the latter relationship is preferred because supercapacitor voltage can be readily measured. To describe the relationship between variable leakage resistance and supercapacitor voltage, piecewise linear approximation is adopted. The start and end points of line segments are determined by the turning points where the resistance-voltage curve has considerable slope change. Since the VLR value only shows a very slow increase when supercapacitor voltage is below 2.6309 V, a constant resistance is used for this voltage range. The constant 173,700 Ω is calculated as the average of the resistances for the voltage range 2.6149-2.6309 V. The variable leakage resistance R_3 is related to supercapacitor terminal voltage V_3 by a piecewise linear function Eq. (21). This approximation matches the calculated values well, as shown in Figure 17.

$$R_3 = \begin{cases} 173700, 0 < V_3 < 2.6309, \\ (-3.906 * V_3 + 10.45) \times 10^6, 2.6309 \leq V_3 < 2.6634, \\ (-1.045 * V_3 + 2.830) \times 10^6, 2.6634 \leq V_3 \leq 2.7000. \end{cases} \tag{21}$$

2.4 Supercapacitor Model Evaluation

This section evaluates the four supercapacitor models (VLR, EIE, two branch model, and three branch model) by comparing the simulated supercapacitor voltages using these models and the measured voltage during three experiments: charging-redistribution, self-discharge, and dynamic charging-discharging experiments. The four model parameter values are first determined.

2.4.1 Supercapacitor Model Parameters

2.4.1.1 VLR Model

The VLR model parameter values of the 10 F sample supercapacitor are shown in Table 1 and Eq. (21). When an external charging or discharging current is present,

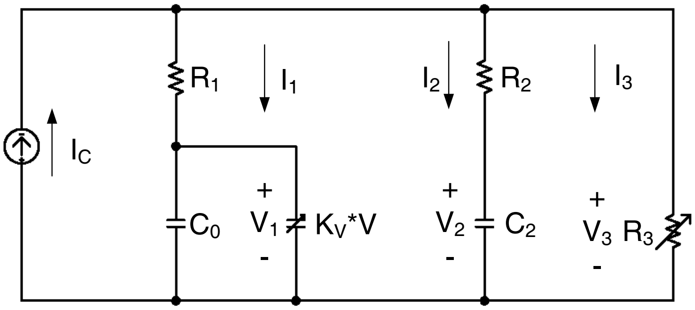


Figure 18: Circuit for charging or discharging analysis.

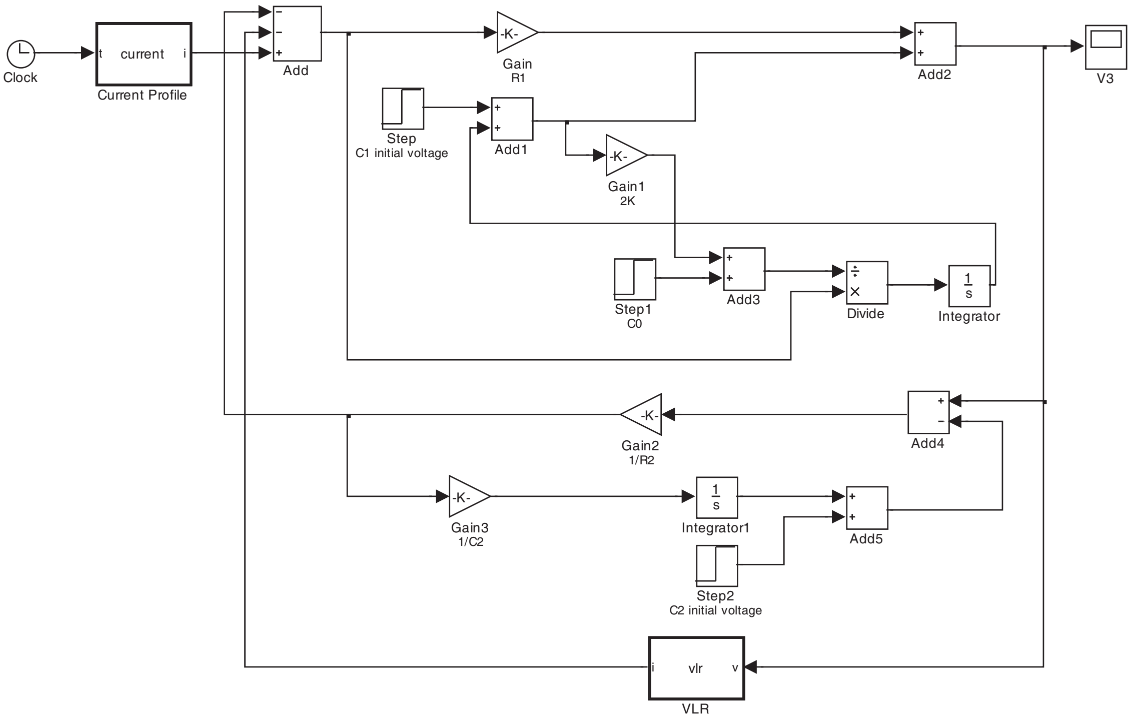


Figure 19: MATLAB Simulink model for VLR model implementation.

as shown in Figure 18, the external current I_C is related to the three branch currents by the following equation:

$$I_C = I_1 + I_2 + I_3. \tag{22}$$

Combining Eqs. (16)-(20) and (22), another MATLAB Simulink model is implemented to determine supercapacitor voltage, as shown in Figure 19. The charging or discharging current I_C generated by the block "Current Profile" is the input. The output is supercapacitor terminal voltage V_3.

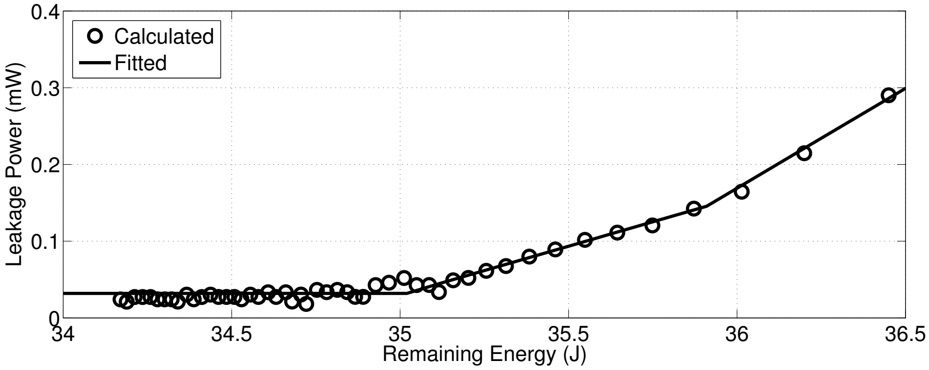


Figure 20: Calculated and fitted supercapacitor leakage powers.

2.4.1.2 EIE Model

The EIE model uses the rated capacitance of a supercapacitor and needs the leakage power profile of a supercapacitor. For the supercapacitor voltage during the self-discharge experiment shown in Figure 9, the leakage power profile is shown in Figure 20. The leakage power profile is approximated using the piecewise linear function Eq. (23). Together with the Eqs. (1)-(3) from Chapter I, the supercapacitor voltage can be calculated.

$$P = \begin{cases} 0.032, 1 \leq E < 35.02, \\ 0.1270 * E - 4.415, 35.02 \leq E < 35.91, \\ 0.2605 * E - 9.209, 35.91 \leq E < 40.00. \end{cases} \tag{23}$$

2.4.1.3 Two Branch Model

The first and second branch parameter values of the two branch model are the same as the VLR model. The parallel leakage resistance is determined using Eq. (24):

$$R_P = \frac{V_N}{I_L}, \tag{24}$$

where V_N is the nominal voltage of a supercapacitor, and I_L is the leakage current of a supercapacitor. For the 10 F sample supercapacitor, $V_N = 2.7$ V and $I_L = 0.03$ mA are given by its datasheet [41]. The parameter values are listed in Table 2.

Table 2: Two Branch Model Parameter Values of a 10 F Supercapacitor

$R_1(\Omega)$	C_0(F)	K_V(F/V)	$R_2(\Omega)$	C_2(F)	$R_P(\Omega)$
0.0677	7.011	1.042	64.52	1.825	90000

Table 3: Three Branch Model Parameter Values of a 10 F Supercapacitor

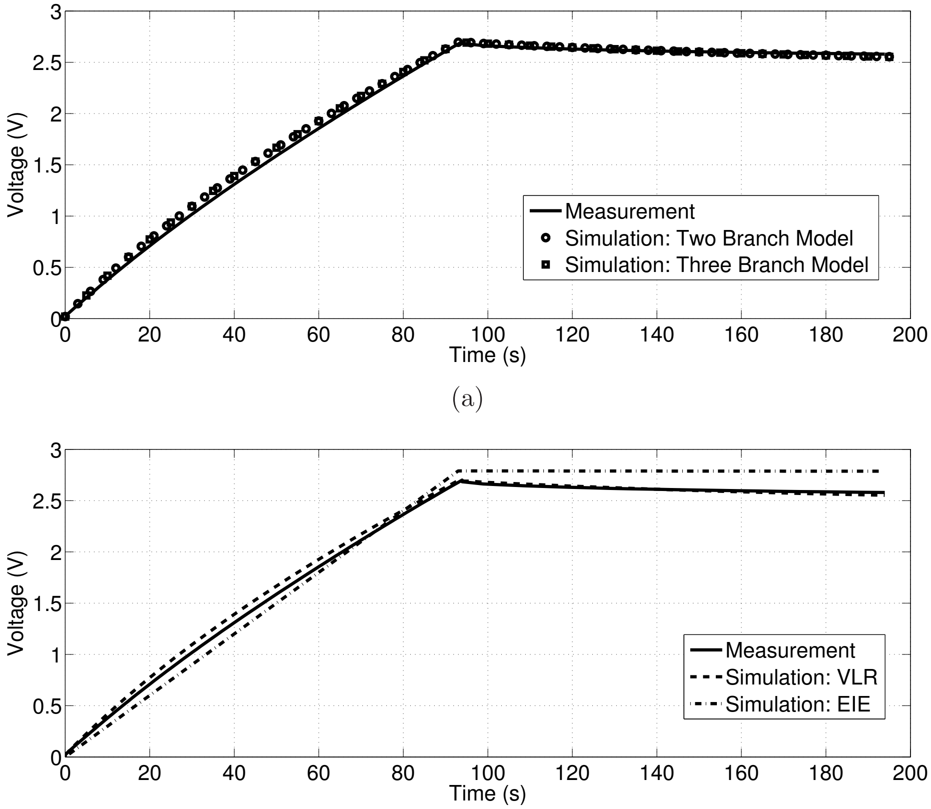
$R_1(\Omega)$	C_0(F)	K_V(F/V)	$R_2(\Omega)$	C_2(F)	$R_r(\Omega)$	C_r(F)	$R_{le}(\Omega)$
0.0677	7.011	1.042	64.52	1.825	11890	0.2005	152470

2.4.1.4 Three Branch Model

The first and second branch parameter values of the three branch model are the same as the VLR model. The other three parameter values R_r, C_r, and R_{le} are determined using the procedure in [17]. The three branch model parameter values are listed in Table 3.

2.4.2 Charging-redistribution Experiment

The four supercapacitor models are first evaluated using a charging-redistribution experiment. The fully depleted 10 F supercapacitor is charged to its rated voltage 2.7 V using a constant current 0.3 A. The charging time is approximately 94 s. When the supercapacitor voltage reaches 2.7 V, the charging current is disconnected and the supercapacitor experiences charge redistribution for the following 100 s. The measured supercapacitor voltage and the simulated voltages using the two branch model and three branch model are shown in Figure 21(a). The measured supercapacitor voltage and the simulated voltages using the VLR model and EIE model are shown in Figure 21(b). For the three equivalent circuit models (VLR, two branch model, and three branch model), the supercapacitor voltage is a zero state response to the charging current. For the EIE model, the consumed energy term in Eq. (2) is zero for the charging phase and the harvested energy term is calculated using the charging current. Both the consumed energy and harvested energy terms are zero for the charge redistribution phase.

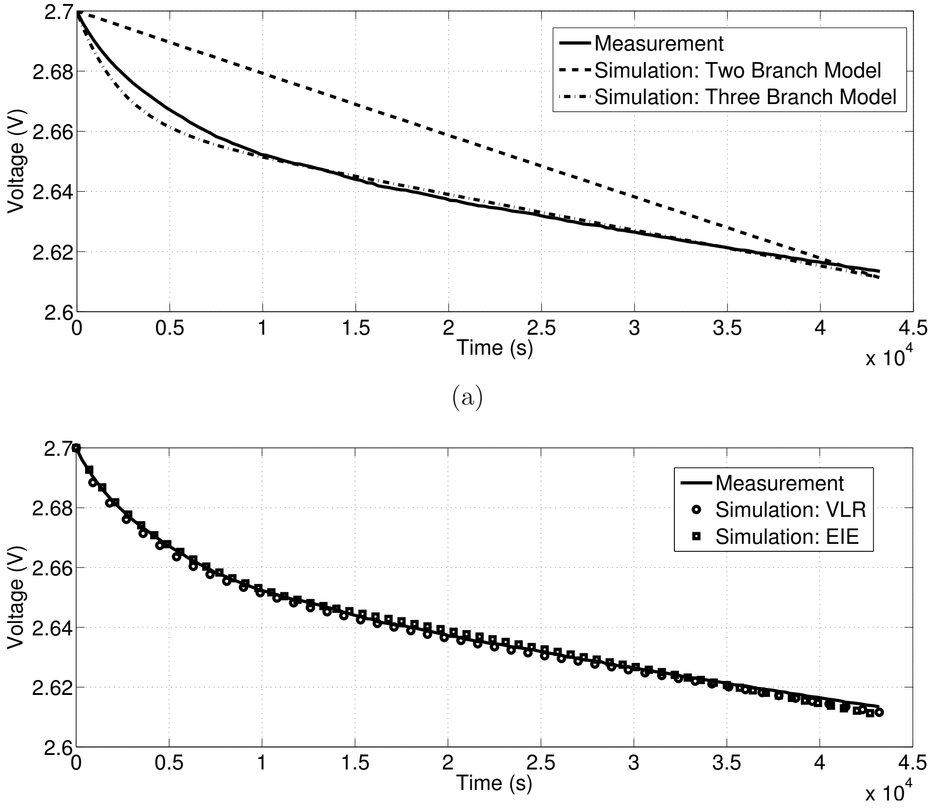


(a)

(b)

Figure 21: Measured and simulated supercapacitor voltages during a charging-redistribution experiment. Charging current is 0.3 A. (a) Two branch model and three branch model. (b) VLR model and EIE model.

As shown in Figure 21, the simulation results using the three equivalent circuit models match the measurement well. However, the EIE model results show noticeable deviations. At the end of the charging phase (94 s), the EIE model predicts a voltage of 2.7898 V, which overestimates the actual voltage. The deviation during the charge redistribution phase is even more significant. The measured supercapacitor voltage is 2.5790 V at the end of this charging-redistribution experiment (194 s). The simulated voltage using the EIE model is 2.7865 V. The deviations are due to the fact that the EIE model does not take into account voltage dependency of capacitance and charge redistribution. The voltage drop during the charge redistribution phase cannot be ascribed to self-discharge.



(a)

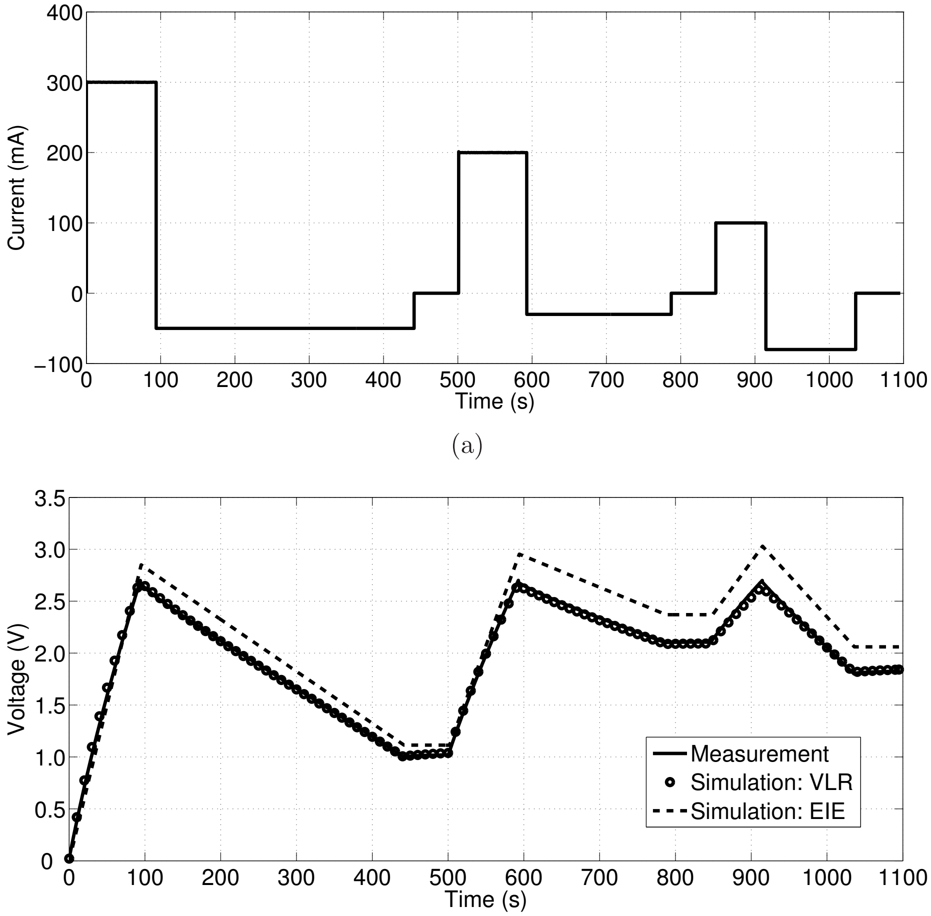
(b)

Figure 22: Measured and simulated supercapacitor voltages during a self-discharge experiment. (a) Two branch model and three branch model. (b) VLR model and EIE model.

2.4.3 Self-discharge Experiment

The four supercapacitor models are then evaluated using the self-discharge experiment shown in Figure 9. Figure 22 shows the measured supercapacitor voltage during this experiment and the simulated voltages. For the three equivalent circuit models, the supercapacitor voltage is a zero input response to the initial state of the circuit where all the capacitances are charged to the rated voltage 2.7 V. In Eq. (2) of the EIE model, both the harvested energy and consumed energy terms are zero.

As shown in Figure 22(a), the simulated voltage using the two branch model has significant error. This observation is consistent with the conclusion that supercapacitor self-discharge cannot be fully characterized using a single exponential function, as



(a)

(b)

Figure 23: A dynamic charging-discharging test. (a) Test current profile. (b) Measured and simulated voltages.

shown in Figure 10. The three branch model simulation results match the measurement well during the last 9 h of this experiment, but underestimate supercapacitor voltage during the first 3 h. Figure 22(b) shows that the simulated voltages using the VLR model and EIE model match the measurement well.

2.4.4 Dynamic Charging-discharging Experiment

The third experiment is a dynamic charging-discharging test. As shown in Figure 23(a), the test current profile is composed of various phases: charging, discharging, and idle. These different phases correspond to the current profiles during various operation modes of an environmentally powered wireless sensor node. The supercapacitor is charged when the energy harvested from the environment is injected into a sensor node. A current is drawn from the supercapacitor if a sensor node is transmitting or receiving data packets. The current is approximately zero when a sensor node is switched to the sleep mode and no harvested energy is injected into the supercapacitor.

In Figure 23(a), the magnitudes of the test currents are determined based on the typical current ranges of solar energy powered wireless sensor nodes. These nodes usually have a solar panel to convert the solar energy into electric energy, an input stage to interface the solar panel and the energy storage system, an energy storage system, an output stage to condition the output voltage of the energy storage system, and a sensor node that consumes energy [25]. The charging and discharging current upper bounds are determined as follows.

Take the sensor node in [8] for example. The output current of the solar panel is around 150 mA. After being conditioned by the maximum power point tracker (MPPT), the current injected into the supercapacitor is around 350 mA. This value is used as the charging current upper bound.

For the output stage, a step-up DC-DC converter LTC3401 [36] is used to boost the supercapacitor voltage. The sensor node is a TmoteSky platform [46]. The upper bound of the discharging current is determined based on the datasheets of LTC3401 and TmoteSky. The major energy consumer of TmoteSky is its radio. The radio only operates if its supply voltage is between 2.1 and 3.6 V. When the radio is receiving packets and the MCU is on, the nominal and maximum current consumptions are 21.8 and 23 mA, respectively. The input and output voltage ranges of LTC3401 are 0.5-5.5 and 2.6-5.5 V, respectively. Considering the rated voltage of the 10 F supercapacitor, the input voltage range of LTC3401 in this dissertation is taken as 1-2.7 V. To determine the upper bound of the current drawn from the supercapacitor, the TmoteSky voltage is assumed to be $V_{TmoteSky} = 3.6$ V and current is $I_{TmoteSky} = 23$ mA. For this current, the power efficiency of LTC3401 is about $\eta = 88\%$. For the supercapacitor voltage range $V_{SC} = 1 - 2.7$ V, the current range is determined as $I_{SC} = 94 - 35$ mA using Eq. (25).

$$\eta = \frac{V_{TmoteSky} I_{TmoteSky}}{V_{SC} I_{SC}} \tag{25}$$

In this dissertation, the upper bound of the discharging current is assumed to be 94 mA. As shown in Figure 23(a), the maximum charging and discharging currents are 300 and 80 mA, respectively, which are smaller than their respective upper bounds.

A comparison between the measured and simulated voltages is shown in Figure 23(b). The simulated voltage using the VLR model matches the measurement well. For the majority of the simulation duration, the EIE simulation results show noticeable deviations from the measurement.

2.5 Summary

This chapter presents an equivalent circuit model called the VLR model for supercapacitors. Supercapacitor physics and three supercapacitor models are reviewed in Section 2.1: EIE model, two branch model, and three branch model. The EIE model is developed based on supercapacitor leakage power profile. This model is accurate in modeling supercapacitor self-discharge. However, it does not consider supercapacitor voltage dependency of capacitance and charge redistribution. The two branch model and three branch model are equivalent circuit models. They take into account the supercapacitor mechanisms that the EIE model does not. However, supercapacitor self-discharge is not well modeled by these two models.

Section 2.2 analyzes the time varying supercapacitor self-discharge and presents the VLR model. The VLR model is an equivalent circuit model that takes into account three mechanisms of supercapacitors: voltage dependency of capacitance, charge redistribution, and self-discharge. The VLR model combines the respective advantages of the EIE model and the two equivalent circuit models.

Section 2.3 illustrates the characterization procedures for the VLR model parameters using a 10 F sample supercapacitor. A MATLAB Simulink model is implemented to determine the variable leakage resistance using the measured supercapacitor voltage during a self-discharge experiment. The variable leakage resistance is related to supercapacitor terminal voltage by a piecewise linear function.

Section 2.4 evaluates the four supercapacitor models by comparing the simulated supercapacitor voltages using these models and the measured voltage during three experiments: charging-redistribution, self-discharge, and dynamic charging-discharging experiments. For the charging-redistribution experiment, the VLR model, two branch model, and three branch model results match the measurement well. The EIE model results show noticeable deviations. For the self-discharge experiment, the VLR model and EIE model are the most accurate. The three branch model results slightly mismatch the measurement during certain period of time. The two branch model is the least accurate for this experiment. For the dynamic charging-discharging experiment, the VLR model is much more accurate than the EIE model.

CHAPTER III

SUPERCAPACITOR CHARGE REDISTRIBUTION AND ENERGY LOSS

Supercapacitor charge redistribution can result in a larger voltage drop than self-discharge. As shown in Figure 9, the voltage drop during the 12 h self-discharge experiment is 0.0849 V. As a comparison, the voltage drop during the charging-redistribution experiment is 0.2887 V, as shown in Figure 12. Supercapacitor terminal voltage is a critical parameter for energy storage system design and power management policy development in environmentally powered wireless sensor nodes. A sensor node can only operate if the supercapacitor voltage is above a certain threshold. Therefore, an analysis of supercapacitor voltage change during charge redistribution is important.

This chapter analyzes supercapacitor voltage change and energy loss during charge redistribution. Section 3.1 investigates the impacts of time and supercapacitor initial state on supercapacitor voltage change during charge redistribution. Section 3.2 explores the relationship between supercapacitor energy loss and time, and the relationship between supercapacitor energy loss and supercapacitor initial state. Section 3.3 is a summary.

3.1 Supercapacitor Charge Redistribution

After a charging or discharging process, the charge stored in a supercapacitor tends to redistribute among different *RC* branches. Circuit theory argues that charge redistribution is a transient response to supercapacitor initial state. The VLR model is used to define supercapacitor initial state. Supercapacitor initial state is characterized by the initial voltages $V_1(0)$ and $V_2(0)$ across the capacitors C_1 and C_2 at time $t = 0$, respectively. The initial time $t = 0$ can be any time instant of interest. The voltages across the capacitors C_1 and C_2 during charge redistribution are V_1 and V_2, respectively. For the sake of simplicity, the initial voltages $V_1(0)$ and $V_2(0)$ are also denoted by V_1 and V_2 in the following sections, respectively. Depending on the previous charging or discharging state of a supercapacitor, the relationship between the initial voltages V_1 and V_2 must fall into one of the three cases: $V_1 = V_2$, $V_1 > V_2$, or $V_1 < V_2$. Charge redistribution takes place when the last two cases are true. The analysis presented in this chapter is based on the second case where $V_1 > V_2$. Similar conclusions hold for the case where $V_1 < V_2$.

3.1.1 Supercapacitor Charge Redistribution Versus Time

As a transient response, charge redistribution is dependent on time and supercapacitor initial state. The effect of time on charge redistribution for a specific supercapacitor initial state is first investigated. Figure 12 shows the measured supercapacitor terminal voltage during a charging-redistribution experiment. In this experiment, the fully discharged 10 F supercapacitor is first charged to 2.7002 V using a 1 A current at 26.515 s. The charging current is then disconnected and the supercapacitor experiences charge redistribution. The supercapacitor terminal voltage decreases during the following period of time. For convenience, Figure 12 is redrawn and shown in Figure 24.

The identification procedure of supercapacitor initial state is illustrated as follows. As already defined, supercapacitor initial state is the initial voltages across the capacitors in the VLR model. What the Maccor testing system and other testers can measure is the supercapacitor terminal voltage. To relate the capacitor voltages to the supercapacitor terminal voltage, the circuit shown in Figure 14 is analyzed.

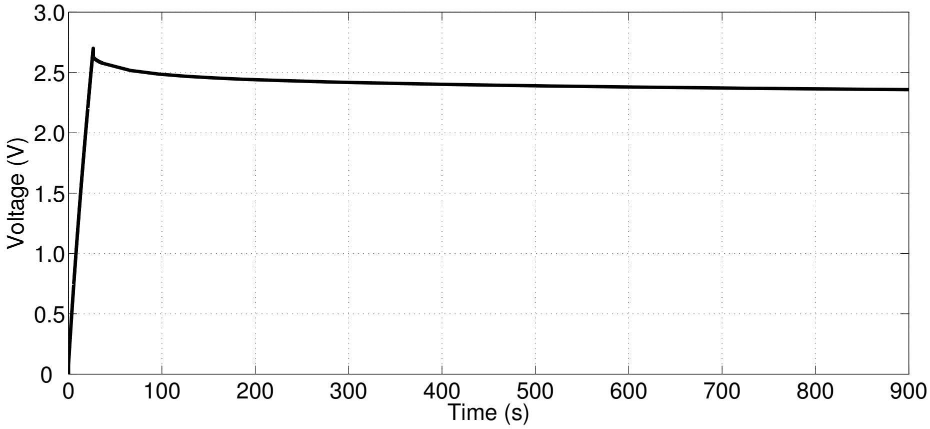


Figure 24: Copy of Figure 12: measured supercapacitor voltage during a charging-redistribution experiment. Charging current is 1 A.

Assuming the current I_1 through R_1 and C_1 flows from the top node to the bottom node, the supercapacitor terminal voltage V_{SC} is related to the C_1 voltage V_1 by Eq. (26). Similarly, V_{SC} is related to the C_2 voltage V_2 by Eq. (27).

$$V_{SC} = I_1 R_1 + V_1 \tag{26}$$

$$V_{SC} = I_2 R_2 + V_2 \tag{27}$$

It can be assumed that $V_1 \approx V_{SC}$ due to the small magnitude of R_1. For instance, the upper bound of the difference between V_{SC} and V_1 during the 1 A charging phase is 0.0677 V with the assumption that the charging current is totally injected into C_1. For smaller charging currents, the difference is smaller. During the charge redistribution phase, the magnitude of I_1 is several mAs and the contribution of R_1 to the terminal voltage is much smaller. Therefore, the measured supercapacitor terminal voltage V_{SC} can be used as the C_1 voltage V_1. On the other hand, V_2 cannot be determined based on V_{SC} since R_2 is large in magnitude. For example, the voltage across R_2 is 0.6452 V if I_2 is 1 mA. The evolution of V_2 can be determined using the VLR model. For the experiment shown in Figure 24, the initial voltages are extracted from the simulation results using the VLR model at $t = 26.52$ s: $V_1 = 2.6527$ V and $V_2 = 0.3176$ V. For accuracy, the initial time is taken as 26.52 s instead of 26.515 s to exclude the voltage drop due to R_1. The 5 ms delay is the time resolution of the Maccor tester. The current I_1 is approximately 1 A (the charging current) at 26.515 s and a couple of mAs (the charge redistribution current) at 26.52 s. As given by Eq. (26), a voltage drop across R_1 is resulted from this current change. At $t = 26.52$ s, the C_1 voltage V_1 (2.6527 V) is very close to the terminal voltage V_{SC} (2.6502 V). On the other hand, at $t = 26.52$ s, the C_2 voltage V_2 (0.3176 V) differs greatly from the terminal voltage V_{SC} (2.6502 V). As a matter of fact, both observations are valid during charge redistribution.

3.1.2 Supercapacitor Charge Redistribution Versus Initial State

Figure 25 shows the effect of supercapacitor initial state on charge redistribution. The fully depleted 10 F supercapacitor is charged to 2.7 V using a constant current source of 1 A and 0.3 A, respectively. The supercapacitor experiences charge redistribution for the following 100 s after the charging current is disconnected. The different charging current results in difference in the supercapacitor initial state of charge redistribution. The initial time for both experiments is the moment at which the charging current is disconnected. Again, to exclude the voltage drop due to R_1, the actual initial time includes a 5 ms delay. For the 1 A current experiment, the initial state is $V_1 = 2.6527$ V and $V_2 = 0.3176$ V at $t = 26.52$ s. The initial state for the 0.3 A case is $V_1 = 2.6869$ V and $V_2 = 0.9280$ V at $t = 93.92$ s. The measured supercapacitor terminal voltage drop is 0.1779 V for the 1 A current experiment and 0.1045 V for the 0.3 A current experiment, respectively. The larger voltage drop in the 1 A current experiment is due to the greater initial voltage difference resulted from a larger charging current.

3.2 Supercapacitor Energy Loss

Depending on the relative magnitudes of V_1 and V_2, the supercapacitor terminal voltage during charge redistribution may change in two ways. If $V_1 > V_2$, part of the

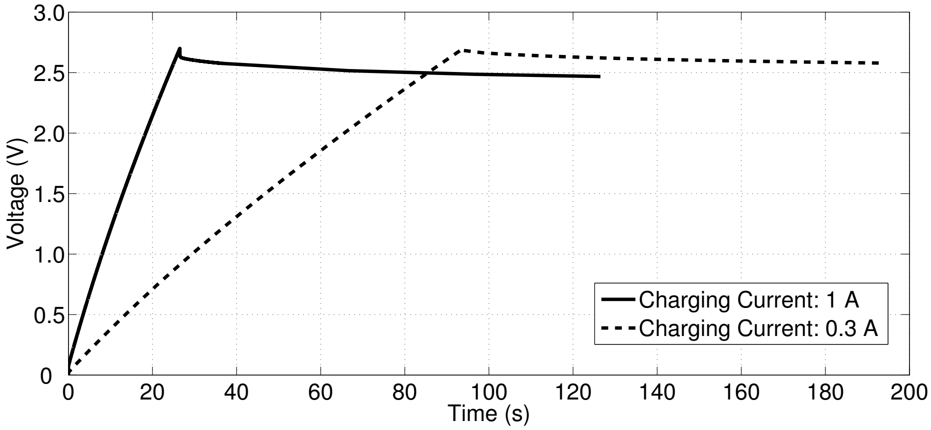


Figure 25: Two charging-redistribution experiments using different charging currents: 1 A and 0.3 A. Redistribution time is 100 s for both experiments.

charge stored in C_1 is redistributed to C_2 and the supercapacitor terminal voltage decreases. If $V_1 < V_2$, charge is transferred from C_2 to C_1 and the supercapacitor terminal voltage increases. For example, Figure 26 shows the VLR component voltages during charge redistribution for initial state: $V_1 = 2.7$ and $V_2 = 2.4$ V. Without loss of generality, the initial state is taken as $V_1 = 2.7$ V and $V_2 = 2.4$ V for the sake of convenience. This initial state is an approximation of a charging process. The simulation results show that if the fully depleted 10 F supercapacitor is charged by a 35 mA current for 880 s, the capacitor voltages will be $V_1 = 2.6917$ V and $V_2 = 2.3972$ V. The current magnitude falls into the typical range of the charging current specified in Section 2.4. Since $V_1 > V_2$, part of the charge stored in C_1 is transferred to C_2. Therefore, V_1 decreases with time and V_2 increases with time. The supercapacitor terminal voltage is equal to the voltage V_3 across R_3, which is approximately equal to the voltage V_1. This is because the resistor R_1 is relatively small.

Charging or discharging currents flow through the resistors during charge redistribution. Therefore, some electric energy is dissipated. Energy loss is the time integral of the dissipated power of a resistor. Because of the changing current and dissipated power, energy loss is also varying with time and dependent on two factors: charge redistribution time and supercapacitor initial state.

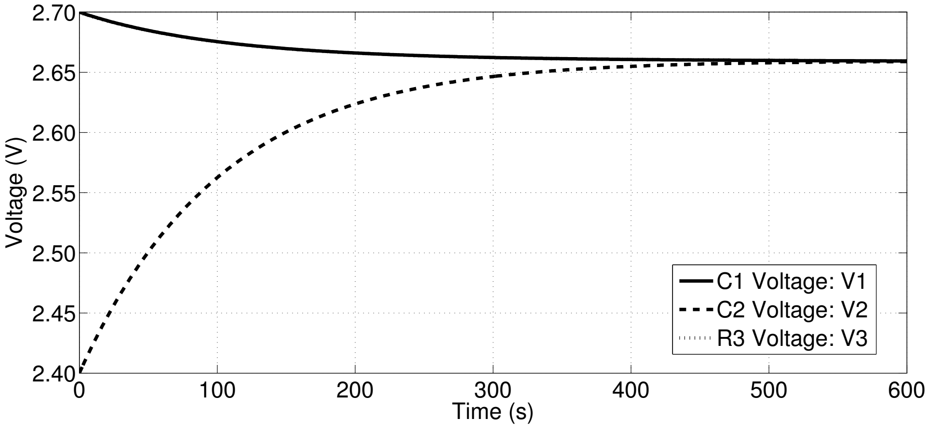


Figure 26: VLR component voltages for initial state: $V_1 = 2.7$ V and $V_2 = 2.4$ V. The curves of V_1 and V_3 almost overlap.

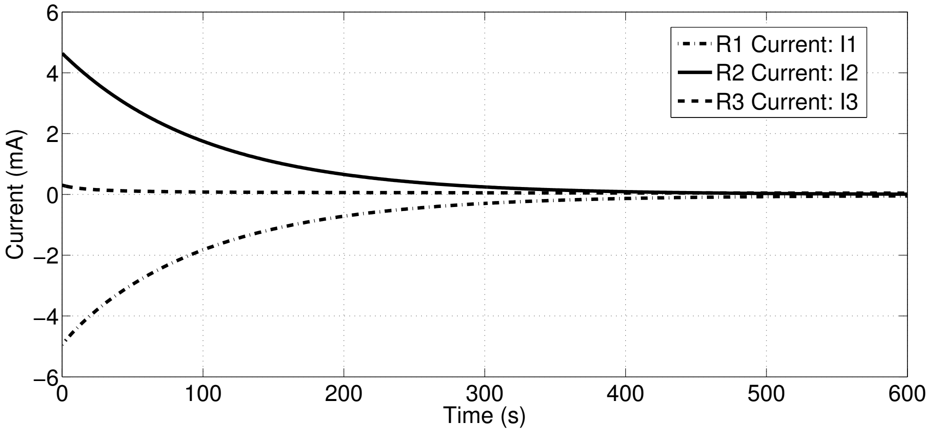


Figure 27: VLR resistor currents for initial state: $V_1 = 2.7$ V and $V_2 = 2.4$ V.

3.2.1 Supercapacitor Energy Loss Versus Time

The relationship between supercapacitor energy loss and charge redistribution time is first investigated while fixing the initial state. Figure 27 shows the currents through the resistors for initial state $V_1 = 2.7$ V and $V_2 = 2.4$ V. The discharging current I_1 through R_1 and charging current I_2 through R_2 are approximately equal in magnitude. Both currents are much greater in magnitude than the current I_3 through R_3.

During charge redistribution, the dissipated power of a resistor is determined using Eq. (28).

$$P = I^2 R \tag{28}$$

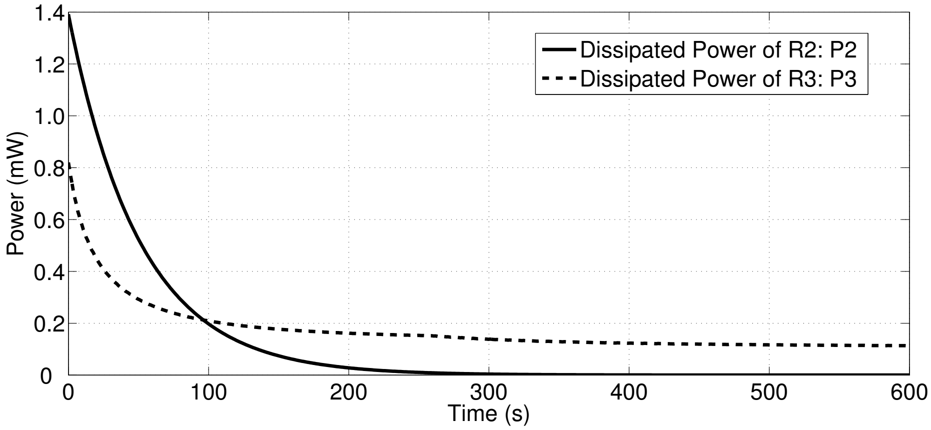


Figure 28: Dissipated powers of R_2 and R_3 for initial state: $V_1 = 2.7$ V and $V_2 = 2.4$ V. Dissipated power of R_1 is negligible.

Figure 28 shows the dissipated powers of R_2 and R_3. The dissipated power of R_1 is negligible given that $|I_1| \approx |I_2|$ and $R_1 << R_2$. The dissipated powers P_2 and P_3 decrease as the charge redistribution time increases because of the decreasing I_2 and I_3. The respective contributions of P_2 and P_3 to the total energy loss vary with time. For the first 100 s, P_2 is greater than P_3. After 100 s, P_3 is greater than P_2. At approximately 300 s, P_2 decays to zero. The charging current I_2 is dependent on the difference between V_1 and V_2: $\Delta V = V_1 - V_2$. As charge redistribution continues, ΔV decreases and I_2 also decreases. The dissipated power P_3 can also be determined by Eq. (29) where $V_3 \approx V_1$.

$$P_3 = \frac{V_3^2}{R_3} \tag{29}$$

During charge redistribution, V_1 decreases and V_3 decreases, too. Therefore, P_3 also decreases as charge redistribution time increases.

The energy loss of a resistor during charge redistribution is calculated using Eq. (30).

$$E(t) = \int_0^t P(t)dt \tag{30}$$

Figure 29 shows the energy losses of R_2 and R_3. The energy loss of R_2 increases with charge redistribution time for the first 300 s, which is approximately three times of the VLR model second branch time constant. After 300 s, E_2 remains constant

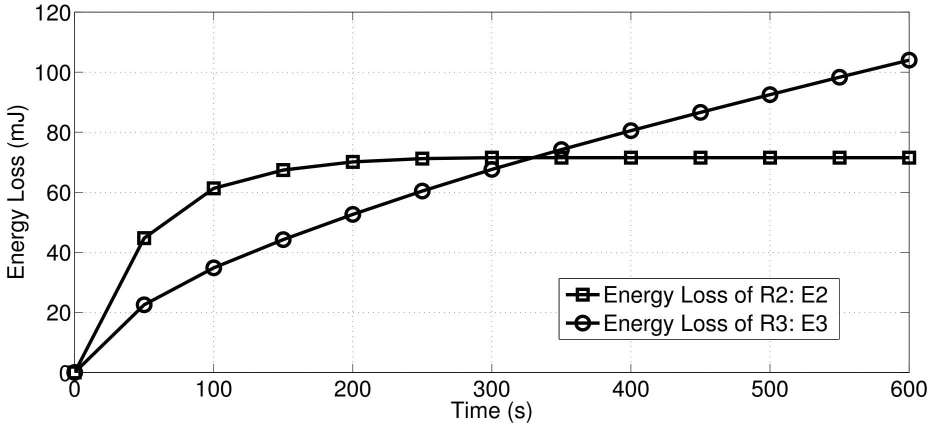
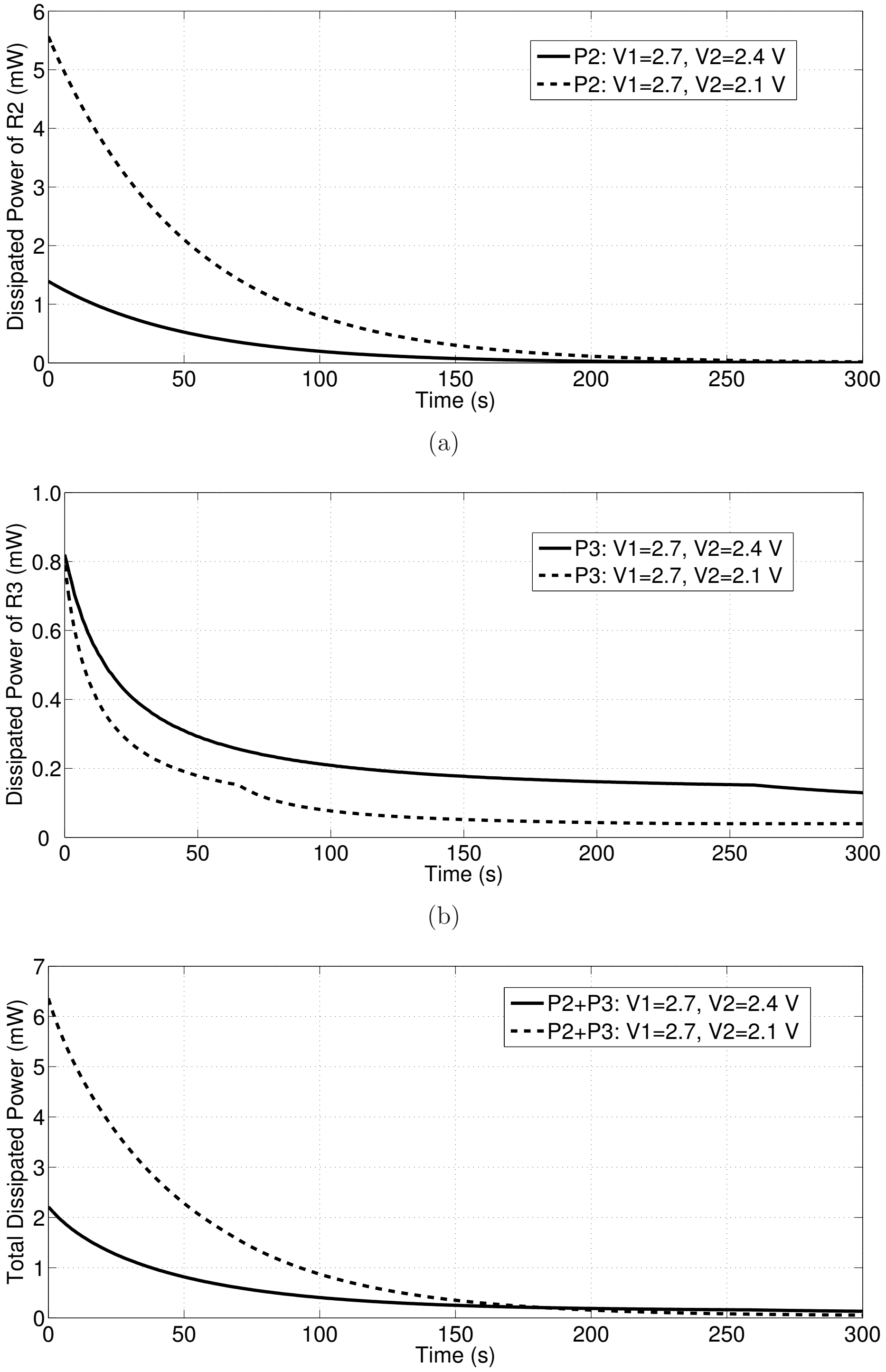


Figure 29: Energy losses of R_2 and R_3 for initial state: $V_1 = 2.7$ V and $V_2 = 2.4$ V.

because P_2 is almost zero. On the other hand, E_3 is always increasing with time. Therefore, the energy loss is mainly contributed by R_2 during the initial phase of charge redistribution and by R_3 for a relatively long term.

3.2.2 Supercapacitor Energy Loss Versus Initial State

The relationship between supercapacitor energy loss and initial state is also studied. The supercapacitor initial state is characterized in two aspects: absolute initial voltages (V_1 and V_2) and relative initial voltage difference ($\Delta V = V_1 - V_2$). The effect of relative initial voltage difference on energy loss is first examined. The energy losses for two relative initial voltage differences are compared. The initial voltage V_1 is fixed at the rated voltage 2.7 V and V_2 is 2.4 V and 2.1 V, respectively. The initial state $V_1 = 2.7$ V and $V_2 = 2.1$ V approximates a charging process where the current is 70 mA and the time is 433 s. The simulation results are $V_1 = 2.6971$ V and $V_2 = 2.0931$ V. Correspondingly, the relative initial voltage differences are 0.3 V and 0.6 V. The dissipated powers of R_2 and R_3 are shown in Figure 30. Again, the dissipated power of R_1 is neglected. As shown in Figure 30(a), for a specific charge redistribution time, P_2 and E_2 increase with the relative initial voltage difference. This is because a greater charging current is generated by a greater initial voltage difference. For R_3, Figure 30(b) shows that P_3 and E_3 decrease with the relative initial voltage difference.

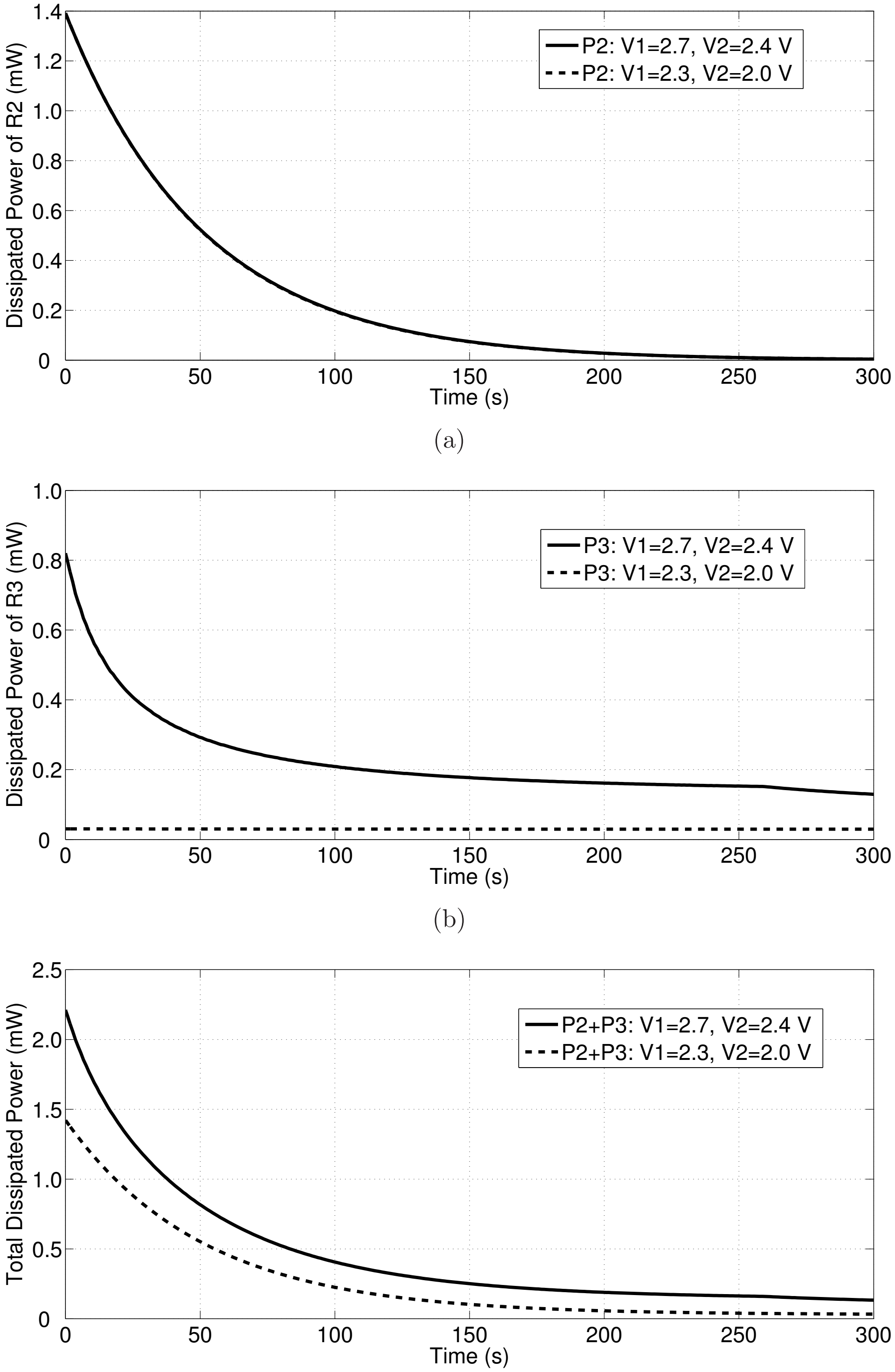


(a)

(b)

(c)

Figure 30: Dissipated powers of R_2 and R_3 for two initial voltage differences: 0.3 V and 0.6 V. The initial states are $V_1 = 2.7, V_2 = 2.4$ V and $V_1 = 2.7, V_2 = 2.1$ V. (a) P_2. (b) P_3. (c) $P_2 + P_3$.



(a)

(b)

(c)

Figure 31: Dissipated powers of R_2 and R_3 for a fixed initial voltage difference: 0.3 V. The initial states are $V_1 = 2.7, V_2 = 2.4$ V and $V_1 = 2.3, V_2 = 2.0$ V. (a) P_2. (b) P_3. (c) $P_2 + P_3$.

This result can be explained by Eq. (29). For a specific charge redistribution time, the voltage V_3 for initial state $V_1 = 2.7$ V and $V_2 = 2.1$ V is less than the voltage V_3 for initial state $V_1 = 2.7$ V and $V_2 = 2.4$ V. Figure 30(c) shows that the total dissipated power $P_2 + P_3$ and the total energy loss $E_2 + E_3$ increase with the relative initial voltage difference for the first 170 s. During this period of time, P_2 dominates the total dissipated power. After 170 s, P_3 dominates the total dissipated power. The total dissipated power $P_2 + P_3$ and the total energy loss $E_2 + E_3$ decrease with the relative initial voltage difference.

The effect of absolute initial voltages on energy loss is also explored. The energy losses for two initial states that have the same relative initial voltage difference are compared. The initial states are $V_1 = 2.7, V_2 = 2.4$ V and $V_1 = 2.3, V_2 = 2.0$ V, respectively. The initial state $V_1 = 2.3$ V and $V_2 = 2.0$ V corresponds to a charging process with a 35 mA current for 722 s. The simulated voltages are $V_1 = 2.3004$ V and $V_2 = 1.9872$ V. The relative initial voltage difference is 0.3 V for both initial states. The dissipated powers of R_2 and R_3 are shown in Figure 31. Figure 31(a) shows that P_2 is almost identical for both initial states due to the same relative initial voltage difference. Figure 31(b) shows that P_3 and E_3 decrease with the absolute initial voltages. Again, Eq. (29) explains this result. The voltage V_3 for initial state $V_1 = 2.3, V_2 = 2.0$ V is less than the voltage V_3 for initial state $V_1 = 2.7, V_2 = 2.4$ V. Figure 31(c) shows that the total dissipated power $P_2 + P_3$ and the total energy loss $E_2 + E_3$ decrease if the absolute initial voltages decrease.

3.3 Summary

This chapter analyzes supercapacitor voltage change and energy loss during charge redistribution. Supercapacitor terminal voltage is a crucial parameter because a wireless sensor node can only operate if the energy storage system provides a voltage above a specific threshold. The voltages across the supercapacitor branch capacitors may be different at a given time instant. Therefore, charge redistribution takes place. Supercapacitor voltage change and energy loss during charge redistribution are analyzed. The VLR model for supercapacitors is used to perform this analysis.

Section 3.1 examines the effects of time and supercapacitor initial state on supercapacitor voltage change during charge redistribution for the case where $V_1 > V_2$. For a fixed supercapacitor initial state, supercapacitor terminal voltage decreases with time because part of the charge stored in C_1 is transferred to C_2. For a particular time, supercapacitor terminal voltage drop is greater if the initial voltage difference $\Delta V = V_1 - V_2$ is larger.

Section 3.2 studies the impacts of time and supercapacitor initial state on supercapacitor energy loss for the case where $V_1 > V_2$. For a fixed supercapacitor initial state, the dissipated powers P_2 and P_3 due to resistors R_2 and R_3 decrease with time. The respective contributions of P_2 and P_3 to the total energy loss vary with time. The energy loss is mainly contributed by R_2 during the initial phase of charge redistribution and by R_3 for a relatively long term. The relationship between supercapacitor energy loss and supercapacitor initial state is also studied. The supercapacitor initial state is characterized in two aspects: absolute initial voltages (V_1 and V_2) and relative initial voltage difference ($\Delta V = V_1 - V_2$). During the initial phase of charge redistribution, the total dissipated power $P_2 + P_3$ and total energy loss $E_2 + E_3$ increase with relative initial voltage difference. For a relatively long term, the total dissipated power $P_2 + P_3$ and the total energy loss $E_2 + E_3$ decrease with relative initial voltage difference. The total dissipated power $P_2 + P_3$ and total energy loss $E_2 + E_3$ decrease if the absolute initial voltages decrease.

CHAPTER IV

EFFECTS OF SUPERCAPACITOR STATE AND ENERGY HARVESTING ON TASK SCHEDULING

Chapter III analyzes the effects of time and supercapacitor initial state on supercapacitor voltage change and energy loss during charge redistribution. To consider the impacts of supercapacitor characteristics on power management in supercapacitor based environmentally powered wireless sensor nodes, the task scheduling problem is studied. A task refers to an event in a wireless sensor node that draws certain amount of energy from the energy storage system. Task scheduling is a fundamental power management problem because the typical operations of a wireless sensor node such as sensing the environment, processing the collected data, and transmitting the data packets can all be taken as different tasks.

Unlike the task scheduling problem in conventional real time systems, it is more complicated in environmentally powered wireless sensor nodes due to the spatial and temporal uncertainty of the harvested energy [38]. It is demonstrated in [44] that greedy scheduling algorithms such as the earliest deadline first (EDF) algorithm may not be suitable for environmentally powered wireless sensor nodes. Therefore, a lazy scheduling algorithm (LSA) is proposed in [44]. In [44], the task scheduling problem is considered for rechargeable battery based systems. This dissertation considers the task scheduling problem in supercapacitor based environmentally powered wireless sensor nodes.

The task scheduling problem is first studied qualitatively. This chapter investigates the effects of supercapacitor state and energy harvesting on task scheduling. Section 4.1 introduces the investigation methodology. Supercapacitor state has three

Table 4: Task Scheduling Simulation Setups and Results

	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6
Supercapacitor State	$V_1 > V_2$	$V_1 < V_2$	$V_1 = V_2$	$V_1 > V_2$	$V_1 < V_2$	$V_1 = V_2$
Energy Harvesting	No	No	No	Yes	Yes	Yes
Task Execution	Greedy	Lazy	N/A	Lazy	Lazy	Lazy
Energy Loss	Greedy	Lazy	Lazy	Greedy	Lazy	N/A

cases: $V_1 > V_2$, $V_1 < V_2$, and $V_1 = V_2$. Energy harvesting is characterized by two scenarios: whether or not energy is harvested during a certain period of time. All together, a set of six cases that covers the various combinations of supercapacitor state and energy harvesting is designed for this investigation. Table 4 lists the simulation setups and results of the six cases. Section 4.2 studies the effects of supercapacitor state on task scheduling. Three cases with different supercapacitor states are examined. No energy harvesting is considered. Section 4.3 explores the effects of energy harvesting on task scheduling. Another three cases with various supercapacitor states are studied. Energy harvesting is considered. Section 4.4 is a summary.

4.1 Methodology

The investigation methodology is illustrated using the simulation setups Case 1 and Case 4. The simulation parameters are selected based on the typical voltage and current ranges presented in Section 2.4.4. Each simulation setup has five components.

First, the supercapacitor state is specified. Supercapacitor state is the initial voltages V_1 and V_2 across the VLR model capacitors C_1 and C_2. Case 1 extracts the supercapacitor initial voltages from an experiment and the other five cases assume initial states with more flexibility for the sake of convenience. For Case 1, the supercapacitor initial state is $V_1 = 1.1855$ V and $V_2 = 0.3994$ V. This initial state is extracted from the simulation results for a charging experiment. The fully discharged supercapacitor is charged by a 110 mA current for 95.5 s to 1.15 V. For Case 4, the supercapacitor initial state is $V_1 = 1.05$ V and $V_2 = 0.50$ V.

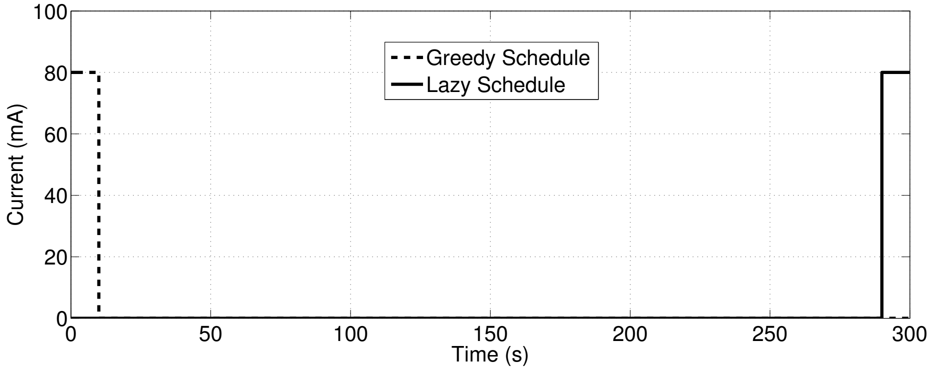


Figure 32: Greedy and lazy schedules of the task.

Second, the task to be scheduled is defined. All the six cases use the same task model. The task to be scheduled is a current pulse, which is defined by four parameters: amplitude 80 mA, release time 0 s, execution time 10 s, and deadline 300 s. The task current magnitude is selected to be smaller than the discharging current upper bound 94 mA specified in Section 2.4.4. The simulation duration is 300 s, which is approximately three times of the second branch time constant of the VLR model.

Third, the greedy and lazy scheduling policies are studied. The supercapacitor only operates if its terminal voltage is greater than a threshold $V_{th} = 1$ V. The supercapacitor threshold voltage and initial voltages fall into the range of 1-2.7 V. The greedy policy schedules the task aggressively. As long as the supercapacitor voltage is above the threshold, the task is scheduled. The greedy schedule is between 0-10 s. The lazy policy works in the opposite way. The task is only scheduled when the deadline is approaching. The lazy schedule is between 290-300 s. The schedules apply to all the six cases. Figure 32 shows the task and the schedules.

Fourth, energy harvesting is considered for Cases 4-6. Cases 1-3 do not consider energy harvesting. As shown in Figure 33, a current pulse is injected into the supercapacitor to model the harvested energy. The current pulse is characterized by its amplitude 100 mA and timing 100-120 s. The current magnitude also falls into the charging current range specified in Section 2.4.4.

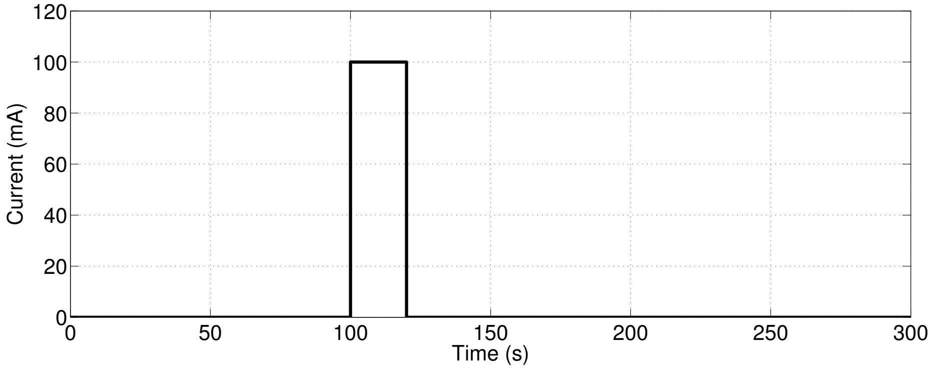


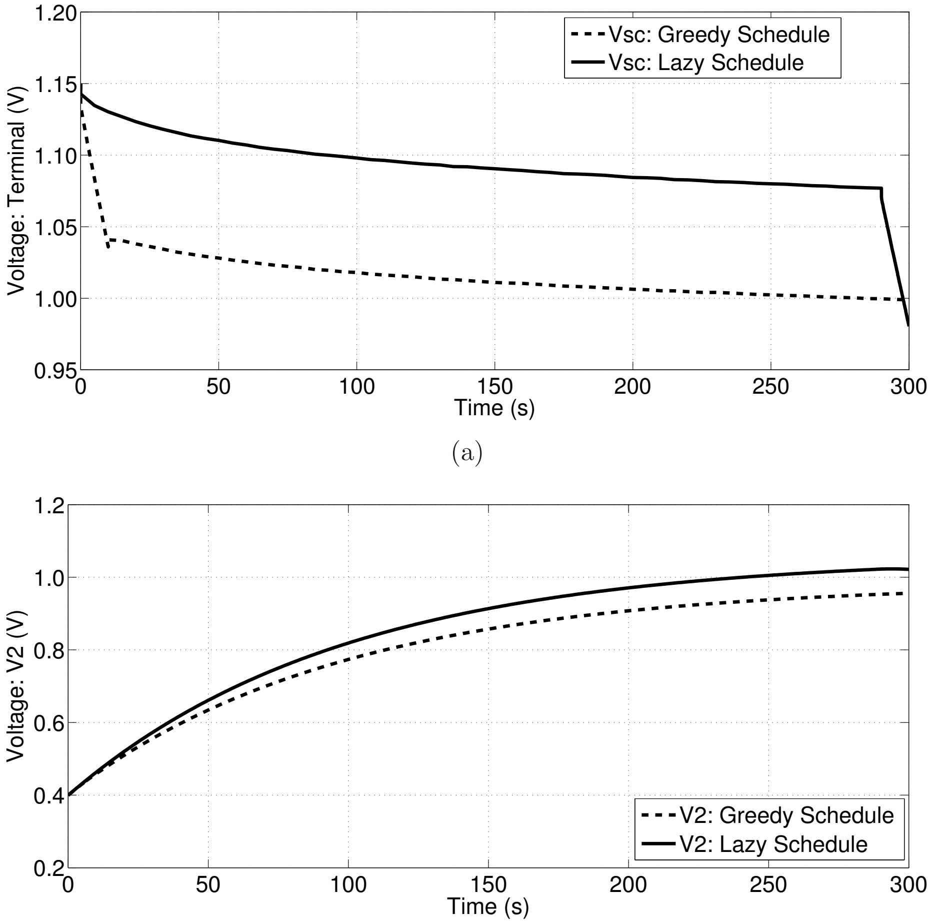
Figure 33: Harvested current pulse for Cases 4-6.

Finally, the scheduling policies are evaluated in terms of two metrics: task execution and energy loss. A successful task execution requires that the supercapacitor terminal voltage is always greater than the threshold while the task is being executed. If the supercpacitor voltage is below the threshold during the task execution period, the execution is considered as unsuccessful. Energy loss is characterized using the approach presented in Chapter III.

4.2 Effects of Supercapacitor State on Task Scheduling

4.2.1 Case 1: $V_1 > V_2$, Without Energy Harvesting

For Case 1, the supercapacitor initial state is $V_1 = 1.1855$ V and $V_2 = 0.3994$ V. No energy harvesting is considered. The supercapacitor terminal voltage is measured experimentally for Case 1. As shown in Figure 34(a), the measured supercapacitor terminal voltage after executing the task is 1.0359 V ($t = 10$ s) and 0.9805 V ($t = 300$ s) for the greedy and lazy schedules, respectively. Therefore, the greedy policy results in a successful execution of the task and the lazy policy an unsuccessful one. The unsuccessful execution using the lazy policy is due to supercapacitor charge redistribution. The task current is mainly drawn from C_1 because the time constant of the first branch is much smaller. Task execution results in a sharp drop in the supercapacitor terminal voltage while the task is being executed. For the greedy schedule, the voltage difference between V_1 and V_2 is reduced and the charge redistribution



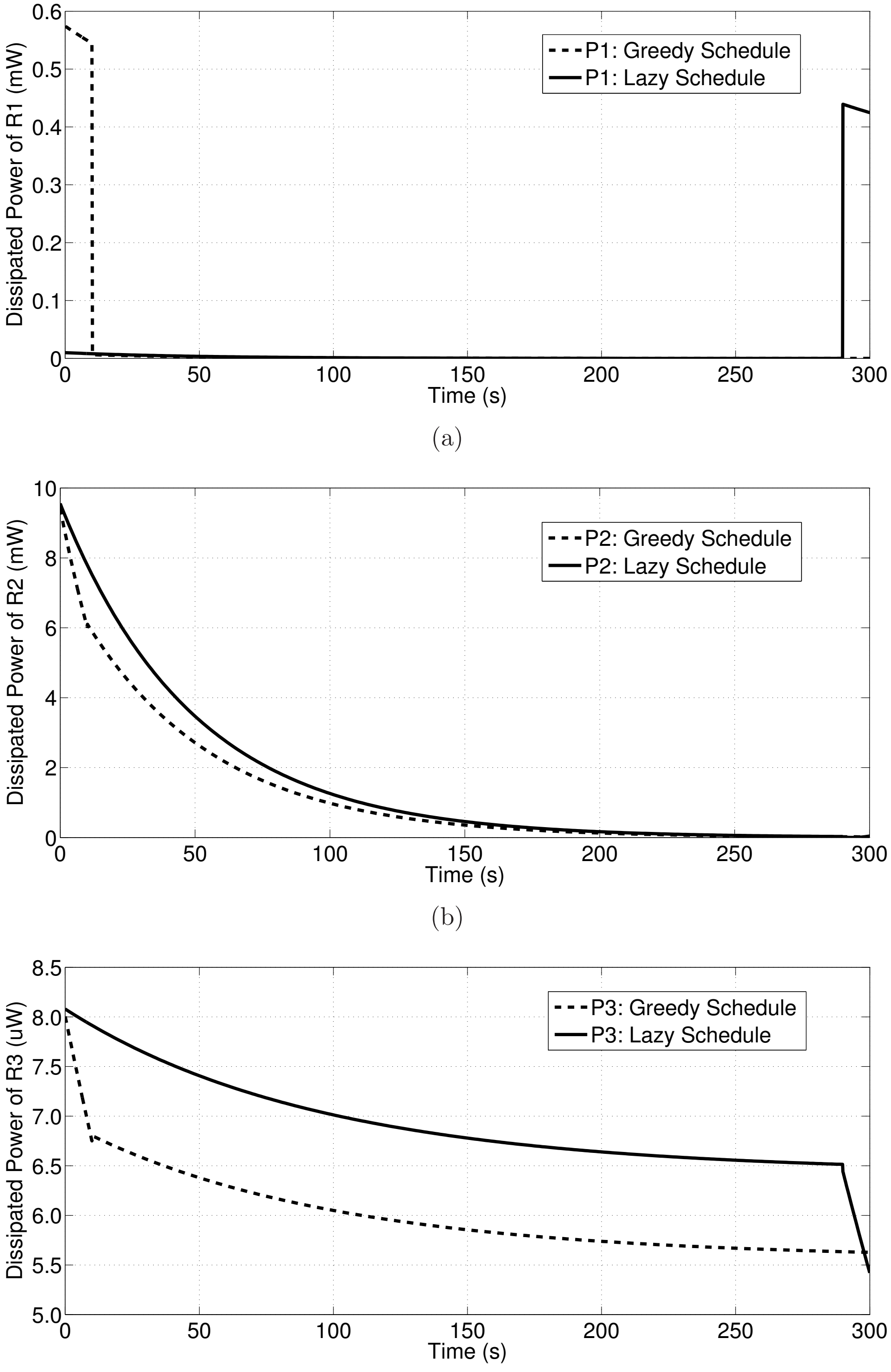
(a)

(b)

Figure 34: Supercapacitor voltages for Case 1. $V_1 = 1.1855$ V and $V_2 = 0.3994$ V. Without energy harvesting. (a) Terminal voltage. (b) V_2.

current is smaller. Therefore, less charge is transferred from C_1 to C_2 and V_2 is lower for the greedy schedule, as shown in Figure 34(b). For the lazy policy, more charge is transferred from C_1 to C_2 during 0-290 s and there is no enough charge stored in C_1 to support the task, which results in an unsuccessful execution of the task.

The dissipated powers of resistors are shown in Figure 35. Figure 35(a) shows that the dissipated power of R_1 needs to be considered when the task is being executed. The task current is mainly drawn from C_1 due to the small time constant of the first branch. When the task is not being executed, P_1 is negligible. As shown in Figure 35(b), the greedy schedule results in a smaller P_2 than the lazy schedule and the

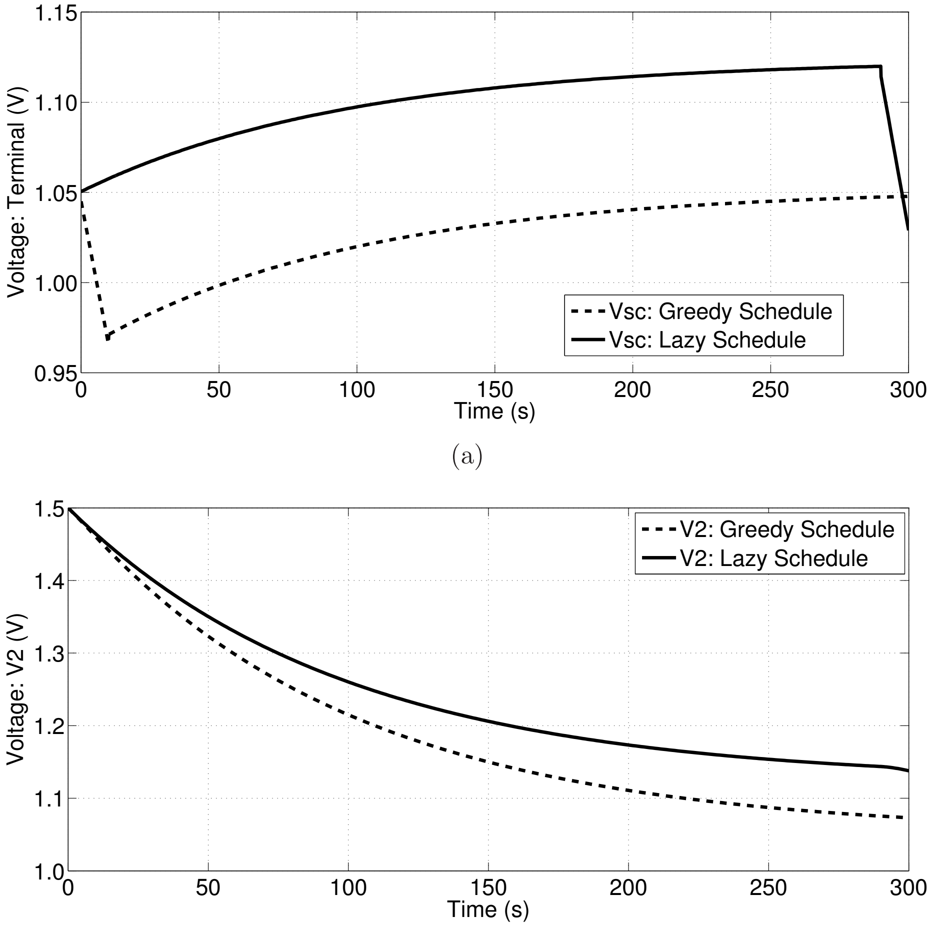


(a)

(b)

(c)

Figure 35: Dissipated powers of resistors for Case 1. (a) P_1. (b) P_2. (c) P_3.



(a)

(b)

Figure 36: Supercapacitor voltages for Case 2. $V_1 = 1.05$ V and $V_2 = 1.50$ V. Without energy harvesting. (a) Terminal voltage. (b) V_2.

energy loss E_2 is also smaller. For the greedy policy, V_1 is reduced after executing the task. The voltage difference $\Delta V = V_1 - V_2$ is also reduced, which results in a smaller dissipated power and a smaller energy loss for R_2. Figure 35(c) shows that P_3 can be neglected. It should be noted that the unit of the dissipated power is μW in Figure 35(c) while Figure 35(a) and Figure 35(b) use mW. Therefore, the greedy scheduling policy is preferred for Case 1 in terms of both metrics.

4.2.2 Case 2: $V_1 < V_2$, Without Energy Harvesting

The initial state of Case 2 is $V_1 = 1.05$ V and $V_2 = 1.50$ V. No energy harvesting is considered. This initial state approximates the supercapacitor voltages after a discharging process. The supercapacitor is discharged by a 60 mA current for 134 s from 1.8 V ($V_1 = V_2 = 1.8$ V). The simulated voltages after discharging are $V_1 = 1.0491$ V and $V_2 = 1.4971$ V.

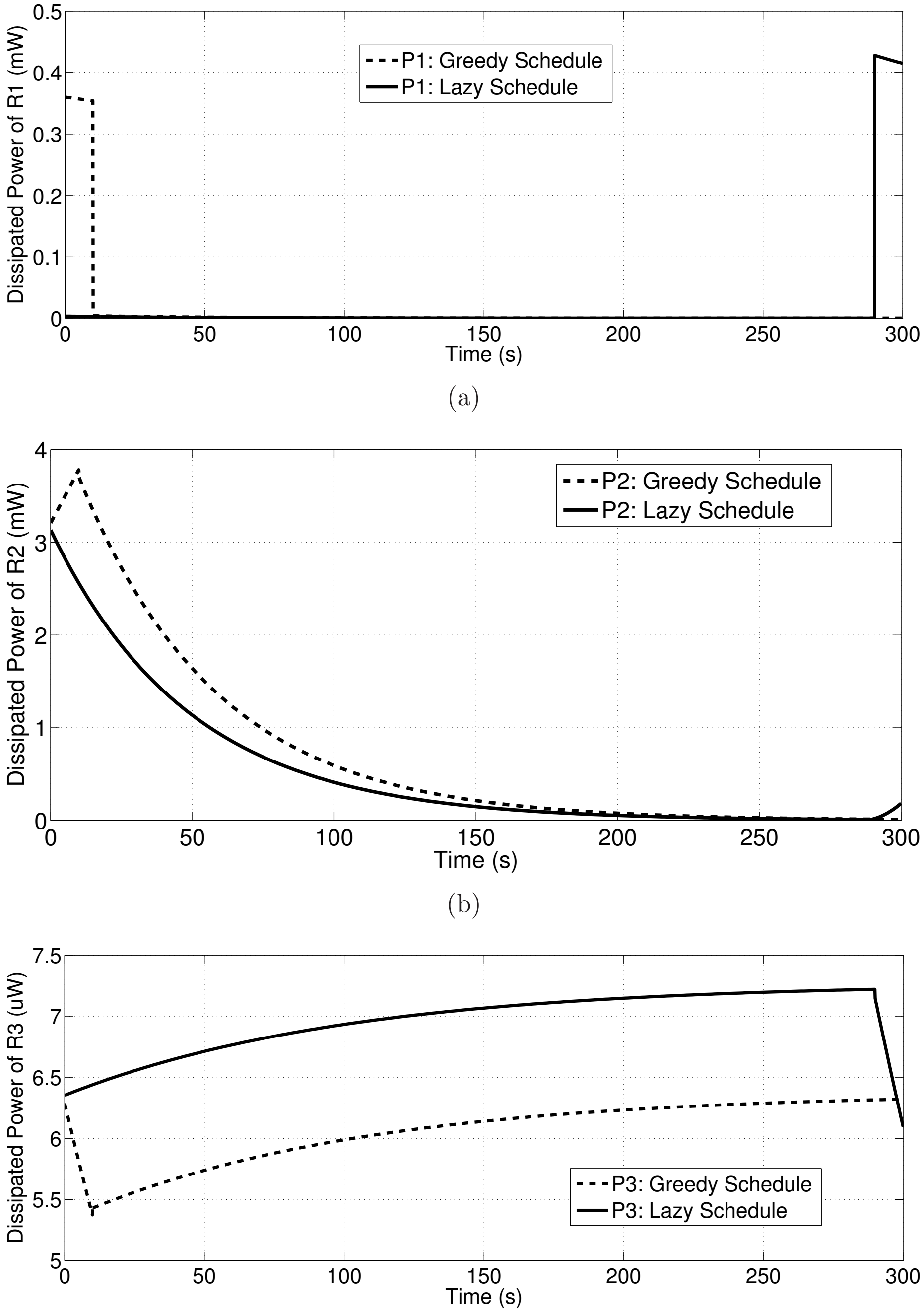
Figure 36(a) shows that the task execution is unsuccessful when the task is scheduled using the greedy policy. On the other hand, the task can be successfully executed if the lazy policy is applied because part of the charge in C_2 is redistributed to C_1. For the greedy schedule, the voltage difference between V_2 and V_1 is increased after the task is executed during 0-10 s, which results in a larger charge redistribution current and a lower V_2, as shown in Figure 36(b).

Figure 37 shows the dissipated powers. It can be observed that P_2 is the major contributor to energy loss. The dissipated power and energy loss for the lazy schedule are smaller. When the greedy policy is adopted, V_1 is reduced after the task is executed. The voltage difference $\Delta V = V_2 - V_1$ increases. Therefore, the dissipated power and energy loss are greater. While Case 1 prefers the greedy schedule, the lazy schedule works better for Case 2 in terms of both metrics.

4.2.3 Case 3: $V_1 = V_2$, Without Energy Harvesting

While supercapacitor charge redistribution takes place when $V_1 \neq V_2$, a supercapacitor may be in the state where $V_1 = V_2$. A charging or discharging action will change this state and result in unequal supercapacitor voltages. For Case 3, the initial voltages are $V_1 = V_2 = 1.05$ V. No energy harvesting is considered.

Figure 38 shows the supercapacitor terminal voltage and V_2. For the greedy schedule, the terminal voltage (which is approximately equal to V_1) has a sharp drop during 0-10 s because of the task current. Therefore, V_2 is greater than V_1 after 10 s and C_2 charges C_1, which explains the increasing terminal voltage and decreasing V_2 during 10-300 s. For the lazy schedule, the terminal voltage and V_2 almost do not change during 0-290 s because of the equal initial voltages. During 290-300 s, the

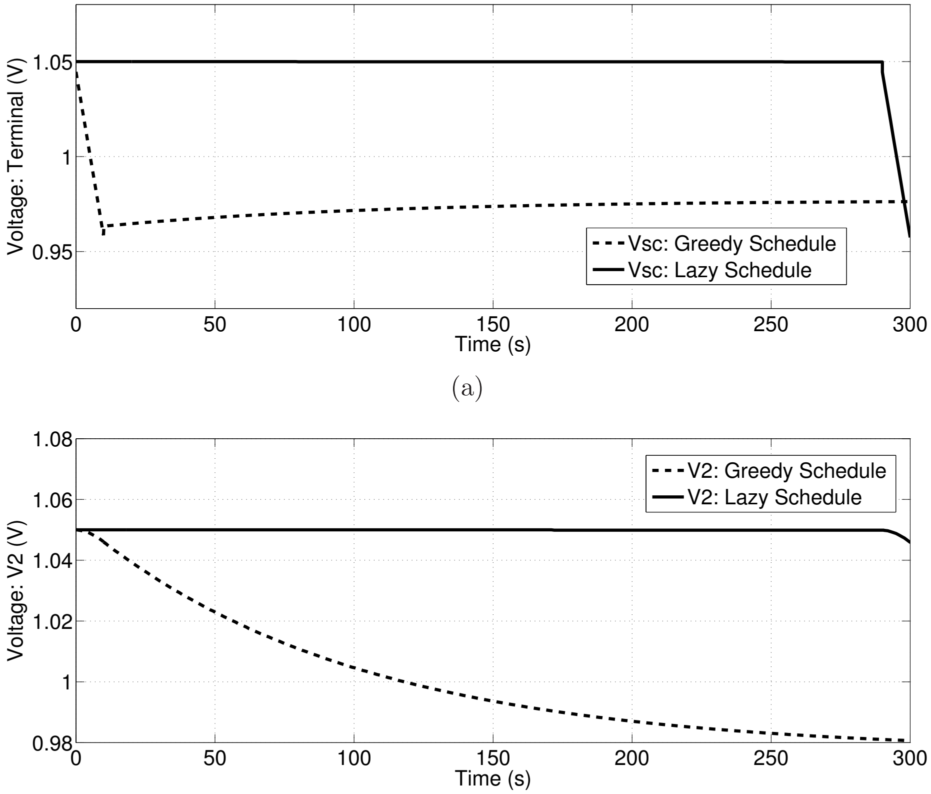


(a)

(b)

(c)

Figure 37: Dissipated powers of resistors for Case 2. (a) P_1. (b) P_2. (c) P_3.



(a)

(b)

Figure 38: Supercapacitor voltages for Case 3. $V_1 = V_2 = 1.05$ V. Without energy harvesting. (a) Terminal voltage. (b) V_2.

task is scheduled. The supercapacitor terminal voltage and V_2 decrease with time. The voltage drop in the terminal voltage is much greater than the voltage drop in V_2 because the task current is mainly drawn from C_1. In terms of supercapacitor terminal voltage, both schedules result in an unsuccessful execution of the task.

The dissipated powers of resistors for both schedules are shown in Figure 39. As shown in Figure 39(a), P_1 is significant when the task is executed for both schedules. The energy losses due to R_1 are almost equal for both schedules. Figure 39(b) shows that the dissipated power P_2 is approximately equal when the task is executed for both schedules. During such periods of time (0-10 s for the greedy schedule and 290-300 for the lazy schedule), the current drawn from C_2 is approximately equal for both schedules. While P_2 is close to zero during 0-290 s for the lazy schedule, the greedy schedule results in P_2 that cannot be neglected because charge redistribution takes place during this period of time. Figure 39(c) shows that the energy loss due to R_3 is greater for the lazy schedule. The greedy schedule reduces the supercapacitor terminal voltage after 10 s and the energy loss is smaller. The total energy loss for the greedy schedule is greater considering the magnitudes of P_2 and P_3. Therefore, the lazy schedule is better in terms of energy loss for Case 3.

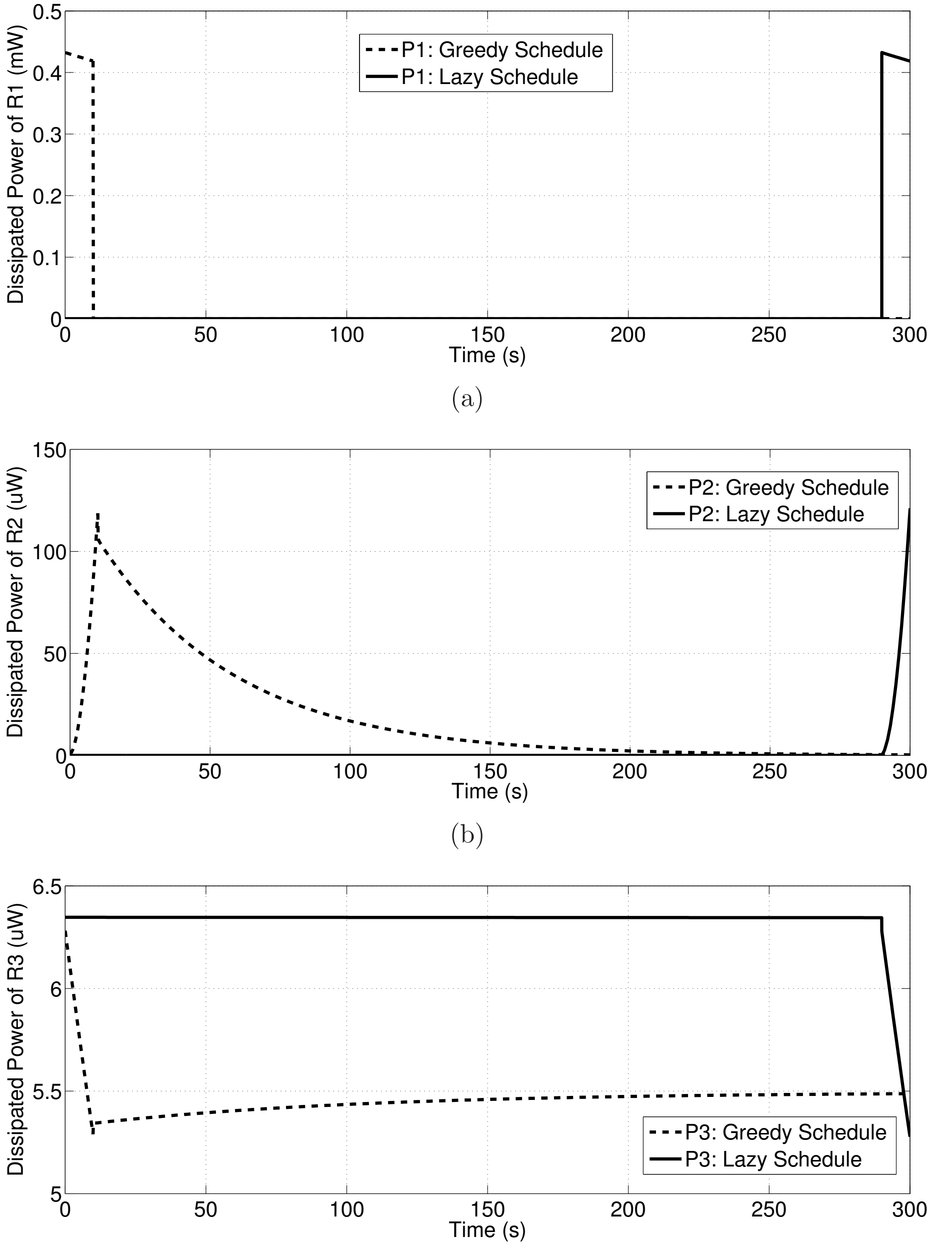
4.3 Effects of Energy Harvesting on Task Scheduling

4.3.1 Case 4: $V_1 > V_2$, With Energy Harvesting

The impacts of energy harvesting on task scheduling can be illustrated by Cases 1 and 4 that have similar supercapacitor initial states. The initial state of Case 4 is $V_1 = 1.05$ V and $V_2 = 0.50$ V. The simulation setup for this initial state is charging the supercapacitor using a 60 mA current for 157 s. The simulated voltages are $V_1 = 1.0500$ V and $V_2 = 0.4981$ V. Energy harvesting is considered. A current pulse is injected into the supercapacitor to model the harvested energy. The current pulse is characterized by its amplitude 100 mA and timing 100-120 s.

As shown in Figure 40(a), the harvested current pulse boosts the supercapacitor terminal voltage during 100-120 s. Consequently, the lazy schedule results in a successful task execution and the harvested current pulse is actually utilized. On the other hand, the greedy schedule does not take advantage of the harvested current pulse. Figure 40(b) shows that V_2 behaves similarly as in Case 1.

Figure 41 shows the dissipated powers. As shown in Figure 41(b), the dissipated power and energy loss for the greedy schedule are smaller. Although both Case 1 and Case 4 favor the greedy schedule for smaller energy loss, the lazy schedule is better for Case 4 to guarantee a successful execution of the task. Therefore, depending on the specific applications the tradeoff between the two performance metrics needs to be considered for Case 4.

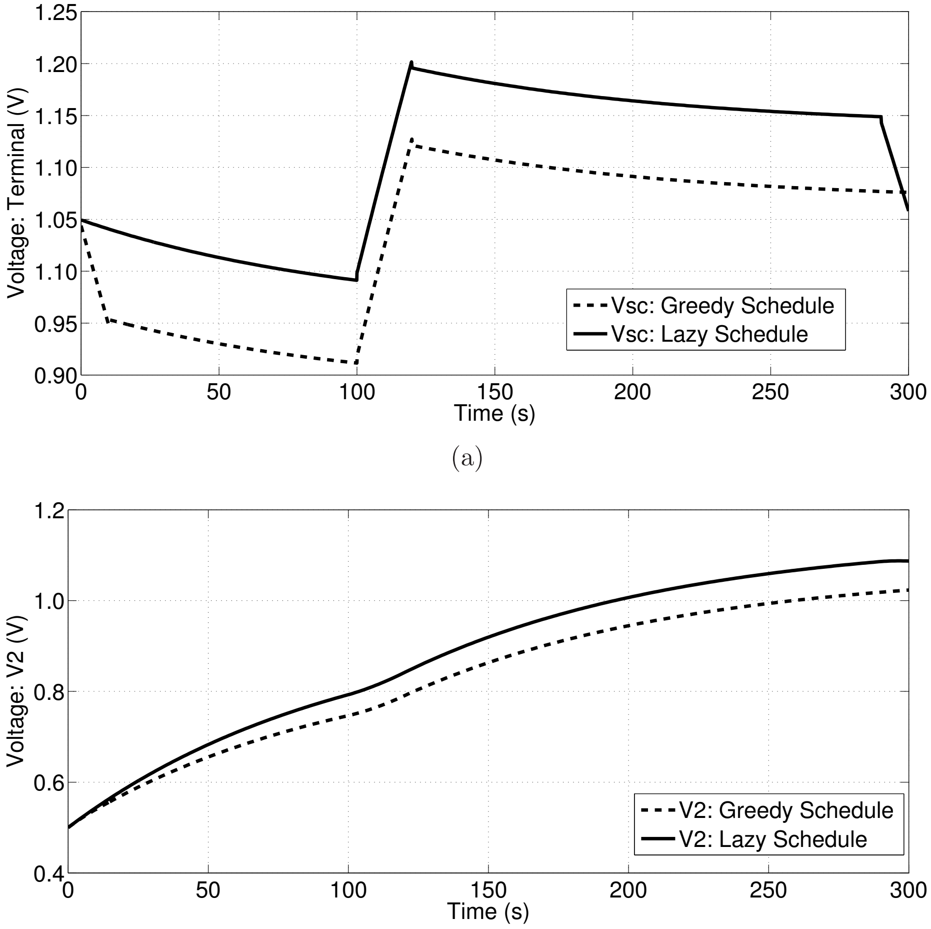


(a)

(b)

(c)

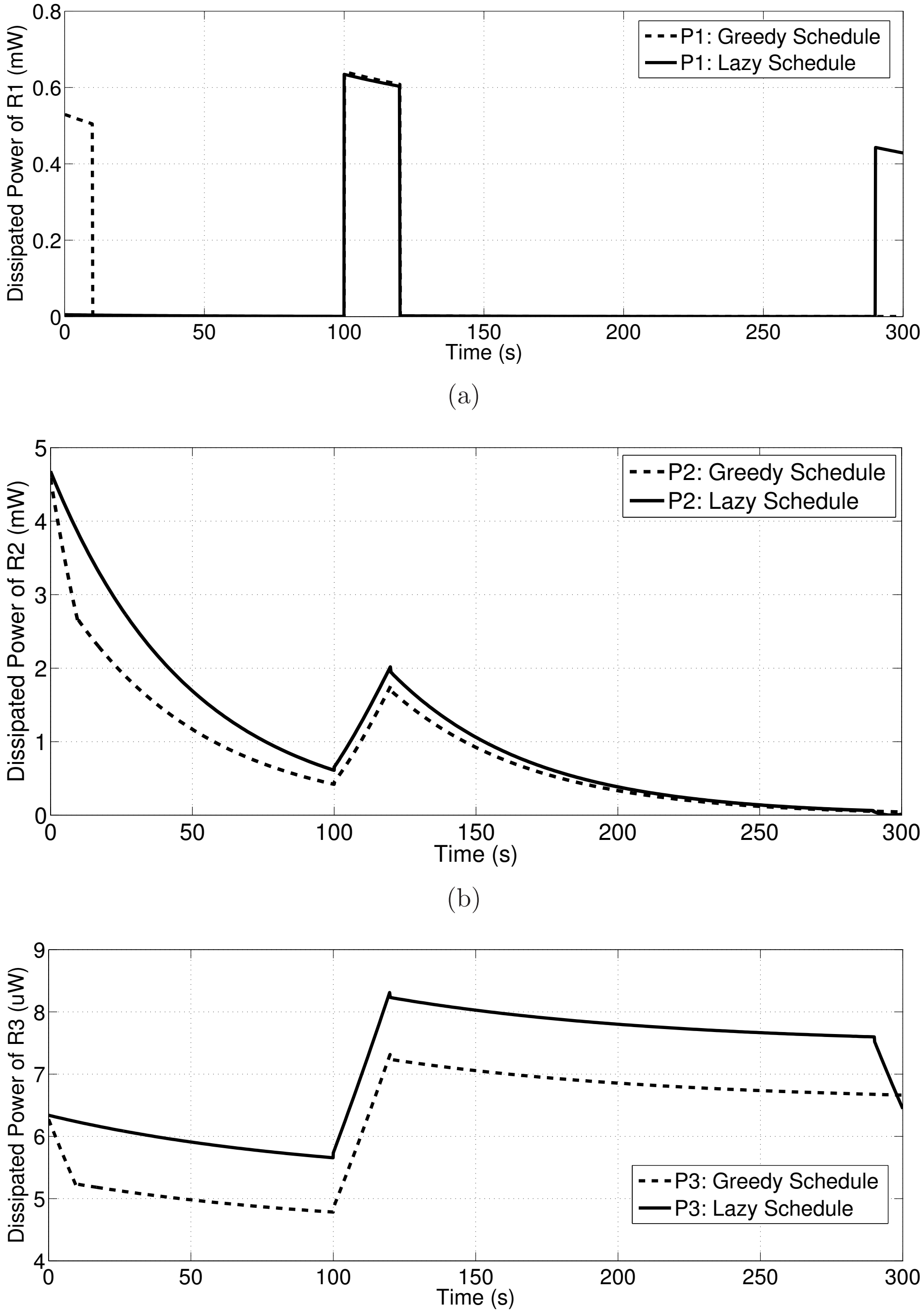
Figure 39: Dissipated powers of resistors for Case 3. (a) P_1. (b) P_2. (c) P_3.



(a)

(b)

Figure 40: Supercapacitor voltages for Case 4. $V_1 = 1.05$ V and $V_2 = 0.50$ V. With energy harvesting. (a) Terminal voltage. (b) V_2.



(a)

(b)

(c)

Figure 41: Dissipated powers of resistors for Case 4. (a) P_1. (b) P_2. (c) P_3.

4.3.2 Case 5: $V_1 < V_2$, With Energy Harvesting

For Case 5, the initial state is $V_1 = 1.05$ V and $V_2 = 1.50$ V and energy harvesting is considered. The intial state is the same as the one specified for Case 2. The lazy policy is preferred in terms of both metrics. Compared with the results shown in Figure 36(a), Figure 42(a) shows that the harvested current pulse results in a greater terminal voltage during 290-300 s for the lazy schedule. Therefore, the task can be successfully executed with a larger voltage margin between the supercapacitor terminal voltage and the threshold voltage.

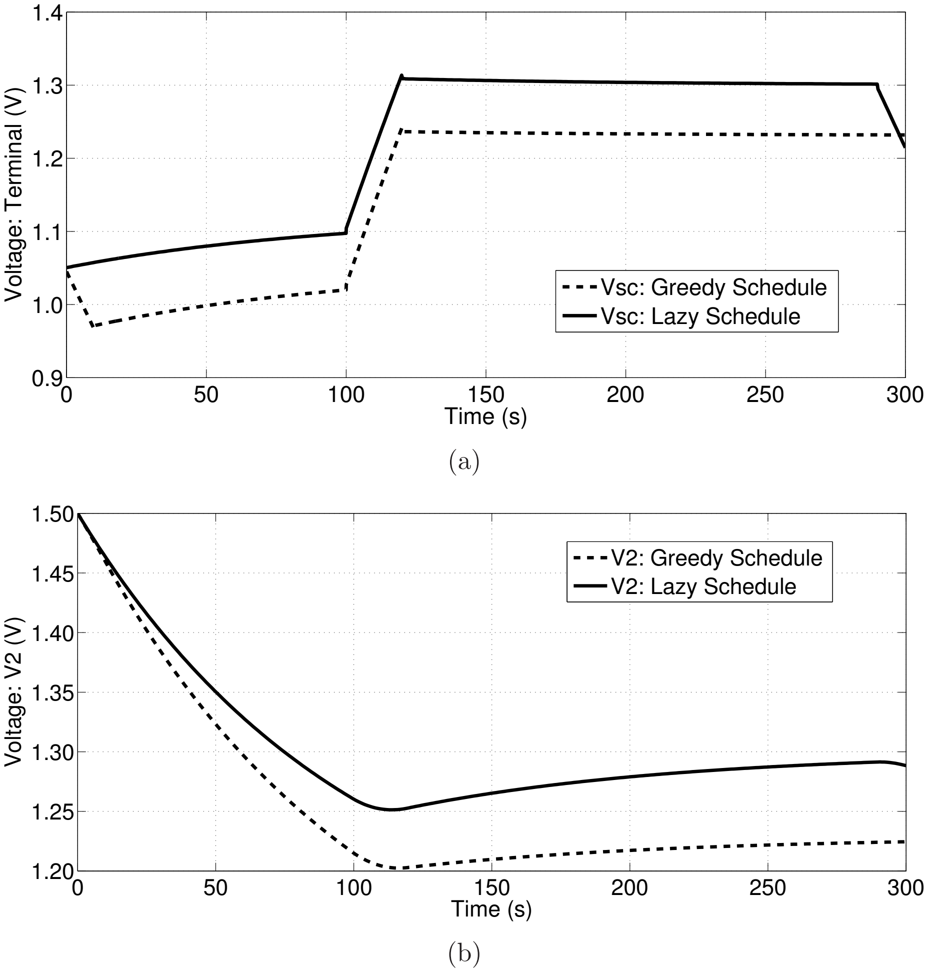
Figure 42(b) shows that V_2 decreases before the harvested current pulse is injected into the supercapacitor for both schedules. During this period of time, V_2 is greater than V_1 and C_2 charges C_1. After 120 s, V_1 is greater than V_2 because the harvested current pulse is mainly injected into C_1. Therefore, charge is transferred from C_1 to C_2. For energy loss, Figure 43 shows that the lazy schedule is better than the greedy schedule. Therefore, the lazy schedule is preferred in terms of both metrics.

4.3.3 Case 6: $V_1 = V_2$, With Energy Harvesting

Case 6 uses the same initial state as the one for Case 3: $V_1 = V_2 = 1.05$ V. Energy harvesting is considered. Figure 44(a) shows that the task execution is unsuccessful when the greedy schedule is used. On the other hand, the lazy schedule takes advantage of the harvested current pulse and results in a successful task execution.

Figure 45 shows the dissipated powers. Both P_1 and P_2 need to be taken into account. For both schedules, P_1 is approximately equal. As shown in Figure 45(b), the greedy schedule has a larger P_2 during 0-100 s because the discharging process during 0-10 s initiates charge redistribution. For the following 200 s, the lazy schedule results in a larger P_2. The total energy loss for both schedules is approximately equal.

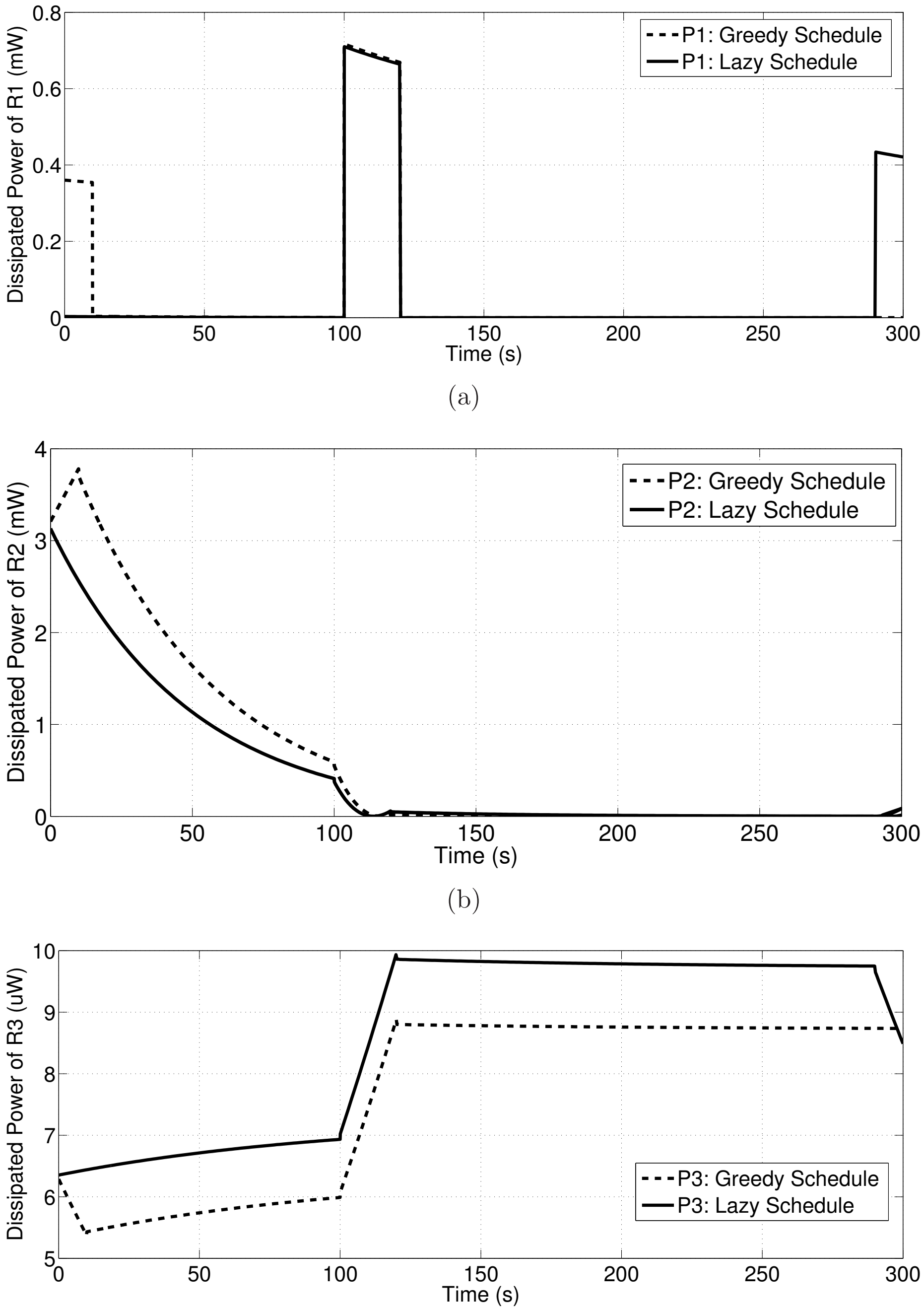
The simulation results of the six cases are summarized in Table 4. The preferred schedule for each case is indicated. For Case 3, both schedules result in an unsuccessful



(a)

(b)

Figure 42: Supercapacitor voltages for Case 5. $V_1 = 1.05$ V and $V_2 = 1.50$ V. With energy harvesting. (a) Terminal voltage. (b) V_2.

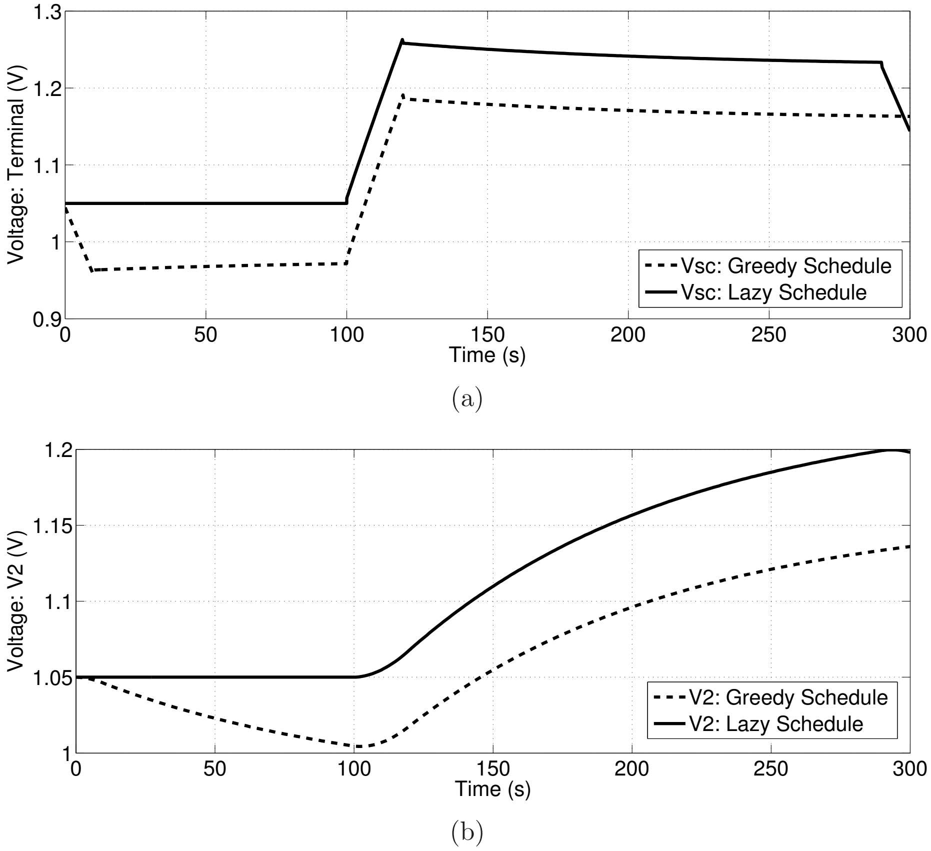


(a)

(b)

(c)

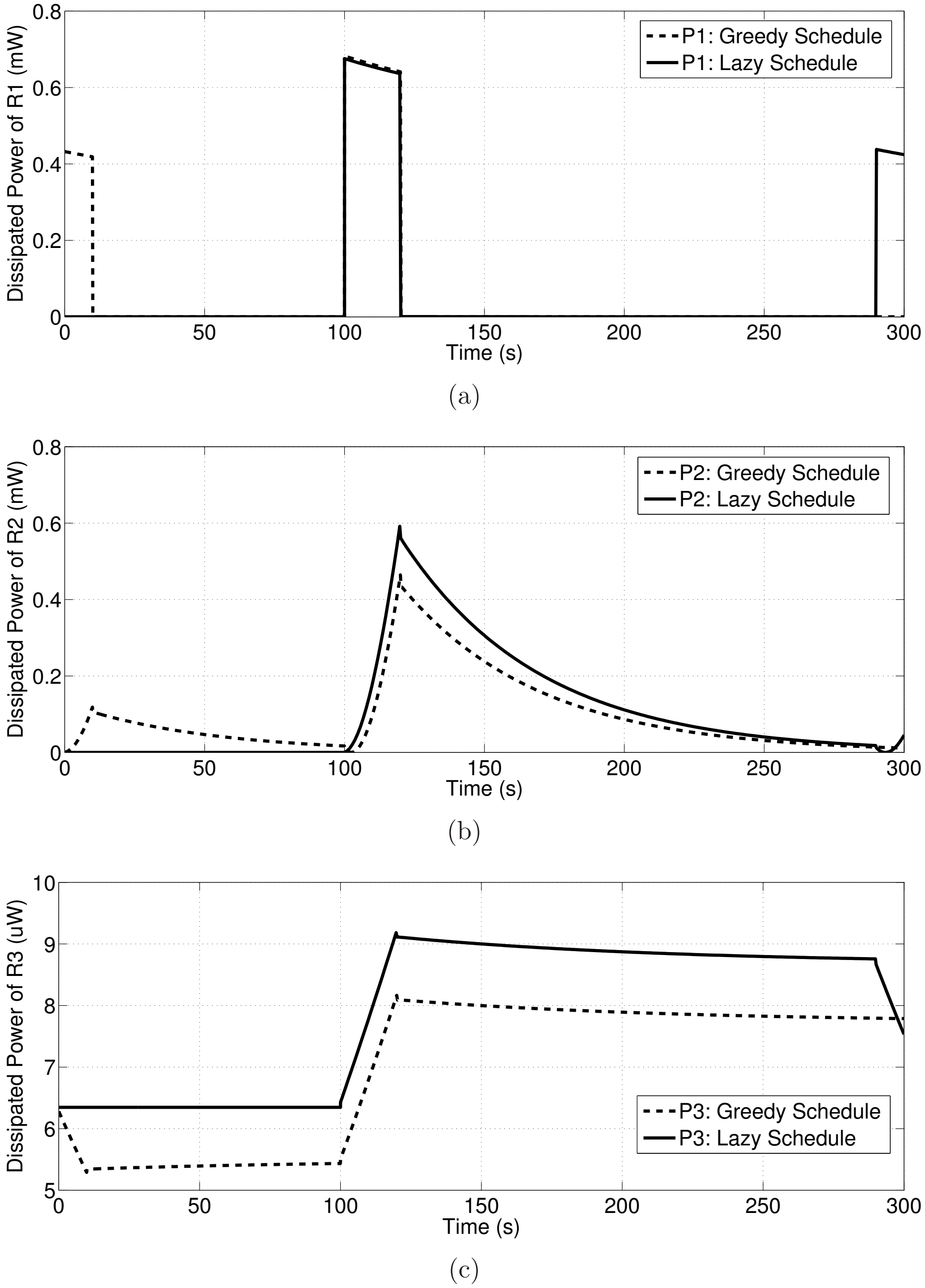
Figure 43: Dissipated powers of resistors for Case 5. (a) P_1. (b) P_2. (c) P_3.



(a)

(b)

Figure 44: Supercapacitor voltages for Case 6. $V_1 = V_2 = 1.05$ V. With energy harvesting. (a) Terminal voltage. (b) V_2.



(a)

(b)

(c)

Figure 45: Dissipated powers of resistors for Case 6. (a) P_1. (b) P_2. (c) P_3.

task execution. There is no difference between the two schedules in terms of this metric and this result is denoted by “N/A”. For Case 6, the greedy and lazy schedules result in approximately equal energy loss. For this metric, the result is also marked as “N/A”.

Based on the results in Table 4, task scheduling rules can be developed. To maximize the number of successfully executed tasks, the rules are: use the greedy schedule when supercapacitor state is $V_1 > V_2$ and energy harvesting is not available; otherwise, use the lazy policy. To minimize energy loss, the rules are: use the greedy schedule when supercapacitor state is $V_1 > V_2$; otherwise, use the lazy policy.

4.4 Summary

This chapter qualitatively studies the task scheduling problem in supercapacitor based environmentally powered wireless sensor nodes. The impacts of supercapacitor state and energy harvesting on task scheduling are investigated. Section 4.1 introduces the investigation methodology. A set of six simulation cases is designed to cover the various scenarios of supercapacitor state and energy harvesting to evaluate the greedy and lazy task scheduling policies in terms of two metrics: task execution and energy loss. Section 4.2 studies the effects of supercapacitor state on task scheduling using the first three simulation cases. Energy harvesting is not considered. Section 4.3 examines the effects of energy harvesting on task scheduling using the last three simulation cases. Energy harvesting is considered. Based on the simulation results, task scheduling rules are developed. To maximize the number of successfully executed tasks, the rules are: use the greedy schedule when supercapacitor state is $V_1 > V_2$ and energy harvesting is not available; otherwise, use the lazy policy. To minimize energy loss, the rules are: use the greedy schedule when supercapacitor state is $V_1 > V_2$; otherwise, use the lazy policy.

CHAPTER V

SCHEDULING INDEPENDENT TASKS

The task scheduling problem is further investigated quantitatively in this chapter and the next chapter. This chapter presents a task scheduling algorithm for nonpreemptable tasks without precedence constraints, which are also called independent tasks. Some operations in a wireless sensor node are independent tasks. For example, a sensor node routes data packets from other sensor nodes. Transmissions of the data packets are independent tasks. The proposed modified earliest deadline first (MEDF) algorithm takes into account supercapacitor state and energy harvesting.

Section 5.1 introduces the system model for a supercapacitor based environmentally powered wireless sensor node. The system model has three components: energy source, energy storage, and energy consumer. Section 5.2 presents the MEDF algorithm. The MEDF algorithm has three steps. First, generate an initial schedule using the earliest deadline first (EDF) algorithm [37]. This step takes care of the timing constraints of tasks. Second, calculate task ready time adjustment margin. This margin determines how much delay is allowed if the initial schedule is adjusted. Third, task ready time offset is determined based on supercapacitor state and energy harvesting. Task start time is the sum of ready time and ready time offset. The MEDF algorithm tries to improve the energy performance of the EDF algorithm while not hurting the timing performance. Section 5.3 implements and evaluates the MEDF algorithm in terms of two metrics: deadline miss rate for timing performance and energy violation rate for energy performance. Section 5.4 is a summary.

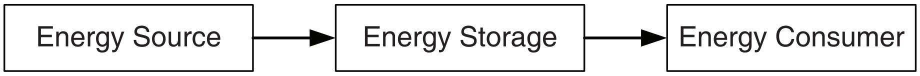


Figure 46: System model for a wireless sensor node.

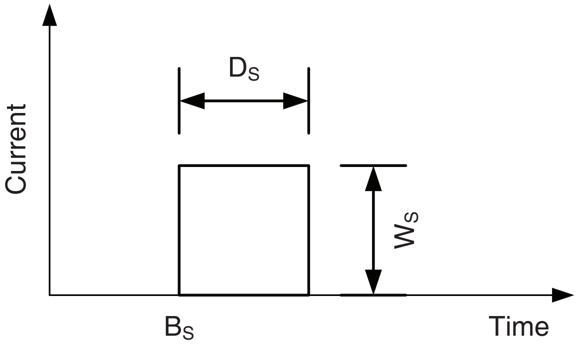


Figure 47: Energy source model.

5.1 System Model

The system model for a supercapacitor based environmentally powered wireless sensor node used in this dissertation to develop task scheduling algorithms is shown in Figure 46. This model has three components: energy source, energy storage, and energy consumer. The component models are elaborated in this section.

5.1.1 Energy Source Model

The energy source model characterizes the harvested energy of a sensor node. The energy source model is composed of multiple current pulses. A single current pulse is shown in Figure 47. This pulse is characterized by three parameters: begin time B_S, duration D_S, and weight W_S. The end time of a pulse is therefore $E_S = B_S + D_S$. Weight W_S is the current magnitude of a pulse. It should be noted that the pulse is the conditioned pulse that is actually injected into the energy storage system. For example, a solar powered sensor node usually has a maximum power point tracker (MPPT) to interface the solar panel and the energy storage system. The current pulse conditioned by the MPPT and fed into the energy storage system is the current pulse used in this model.

5.1.2 Energy Storage Model

The energy storage system is a single supercapacitor. An environmentally powered wireless sensor node usually has various power conditioning components such as MPPT and DC-DC converters. In this system model, the effects of interfacing circuits are considered by the energy source and energy consumer models. The VLR model developed for the 10 F supercapacitor in Chapter II is used in this chapter as the energy storage model. Supercapacitor initial state is characterized by the initial voltages across the capacitors in the VLR model.

5.1.3 Energy Consumer Model

The energy consumer is modeled as a set of tasks that draw certain amount of current from the supercapacitor. As shown in Figure 48, each task is defined by four parameters: release time R, execution time E, absolute deadline D, and weight W. The release time of a task is the instant of time at which the task becomes available for execution. The execution time is the amount of time required to complete the execution. The absolute deadline is the instant of time by which the task execution is required to be completed. There is another deadline called the relative deadline, which is defined as the length of time from the release time to the time instant when the task execution completes. The relative deadline in then determined as $D - R$. In this chapter, the term "deadline" refers to absolute deadline by default. The release time, execution time, and deadline are the temporal parameters defining the timing constraint of a task. The weight of a task is its current magnitude. This parameter defines the energy constraint of a task.

In this chapter, the tasks are independent and nonpreemptable. Independent tasks can be executed in any order. In other words, the tasks do not have precedence constraints. The tasks are also nonpreemptable. A nonpreemptable task must be executed from start to completion without interruption.

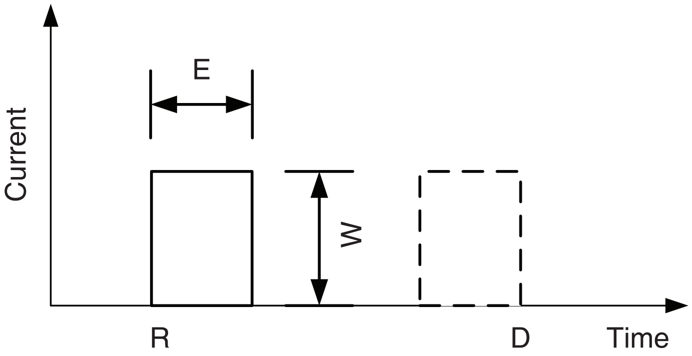


Figure 48: Energy consumer model.

5.2 Modified Earliest Deadline First (MEDF) Algorithm

The MEDF algorithm has three steps. First, an initial schedule is created using the EDF algorithm. The initial schedule only considers the timing constraints of tasks. The initial schedule determines the ready time of each task. The ready time of a task is the earliest time instant at which the task can be executed. Second, the ready time adjustment margin is calculated based on the initial schedule. As shown in Chapter IV, a task execution may be postponed to ensure that the supercapacitor voltage is above the voltage threshold. If a task has some delay margin to guarantee that its deadline can be still respected, it may be preferable to take advantage of this margin to take care of the energy constraint. This step determines the maximum available delay margin. Third, the ready time offset is determined based on supercapacitor state and energy harvesting. The start time of a task is its ready time plus the ready time offset. The MEDF schedule is then finalized.

5.2.1 Generate An Initial Schedule Using EDF Algorithm

The EDF algorithm is used to create an initial schedule for a set of N tasks. Each task T_i is defined by four parameters (R_i, E_i, D_i, W_i). The EDF algorithm schedules tasks based on their deadlines. The task with the earliest deadline is first scheduled. The algorithm first sorts all the ready but not scheduled tasks in the ascending order of their deadlines. An intermediate variable called current time T_C is introduced, which is initialized as $T_C = 0$ at the beginning of the algorithm. The ready time of each task A_i is calculated as the maximum value between the current time T_C and its release time R_i. Once the ready time of a task is determined, the current time is updated as the end time of the current task: $T_C = A_i + E_i$. The initial schedule T_{EDF} is defined by the ready time of each task A_i: $T_{EDF} = T_i(A_i, E_i, D_i, W_i)$. The algorithm is shown in Algorithm 1.

Algorithm 1 Generate An Initial Schedule Using EDF Algorithm

```
Require: A set of $N$ ready but not scheduled tasks: $T = T_i(R_i, E_i, D_i, W_i)$.
Sort $N$ tasks in the ascending order of their deadlines.
Current Time $T_C = 0$                                  ▷ Initialization.
for $i = 1 : N$ do
    Ready Time $A_i = max(T_C, R_i)$               ▷ Tasks are nonpreemptable.
    Current Time $T_C = A_i + E_i$                       ▷ Update $T_C$.
end for
Algorithm output is initial schedule $T_{EDF}$ defined by task ready time $A_i$: $T_{EDF} = T_i(A_i, E_i, D_i, W_i)$.
```

5.2.2 Calculate Ready Time Adjustment Margin

The initial schedule is a greedy schedule: as long as the task is ready, it will be executed. As shown in Chapter IV, if supercapacitor voltage falls below the operation threshold during the execution of a task, the task execution is unsuccessful even though its deadline is respected. If there is some delay margin allowing the task execution to be postponed to some extent such that its deadline is still respected and supercapacitor operation requirement is met at the same time, the adjusted schedule is then a better choice. This subsection examines how much the delay margin can be for a task scheduled using the EDF algorithm.

The fundamental rule to determine the delay margin is to improve the energy performance while not affecting the timing performance of the initial schedule. Several intermediate timing parameters are defined to determine the delay margin. Figure 49 shows the definitions of two parameters: maximum delay margin MM and ready

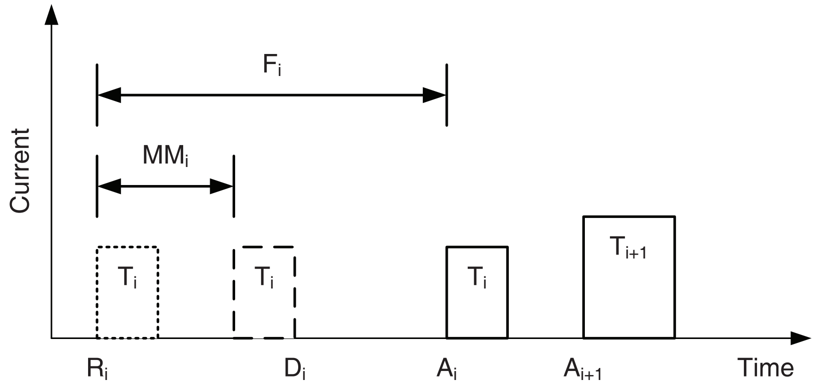


Figure 49: Definitions of maximum delay margin and ready time flag.

time flag F for task T_i. Figure 49 also shows the initial schedule of two tasks T_i and T_{i+1} denoted by the solid line rectangles. For task T_i, the earliest and latest possible schedules that respect its deadline are represented by the dotted and dashed line rectangles, respectively.

The maximum delay margin is defined as the maximum length of time by which a task can be postponed for execution from its release time so that the deadline is not missed. The maximum delay margin is the difference between relative deadline and execution time, as given by Eq. (31). The maximum delay margin is a property of a task itself and is not dependent on any other tasks or the specific schedule.

$$MM = D - R - E \tag{31}$$

The ready time flag is defined as the difference between the ready time and the release time, as given by Eq. (32). This parameter depends on the specific schedule of a task.

$$F = A - R \tag{32}$$

Depending on the relative magnitudes of maximum delay margin and ready time flag, the ready time of a task may be or may not be adjusted. Specifically, if $F > MM$, as shown in Figure 49, the deadline of a task is already missed. Therefore, there is no need to consider adjusting its ready time. On the other hand, if $0 \leq F \leq MM$, there may be some delay margin a task can take advantage of. The case where

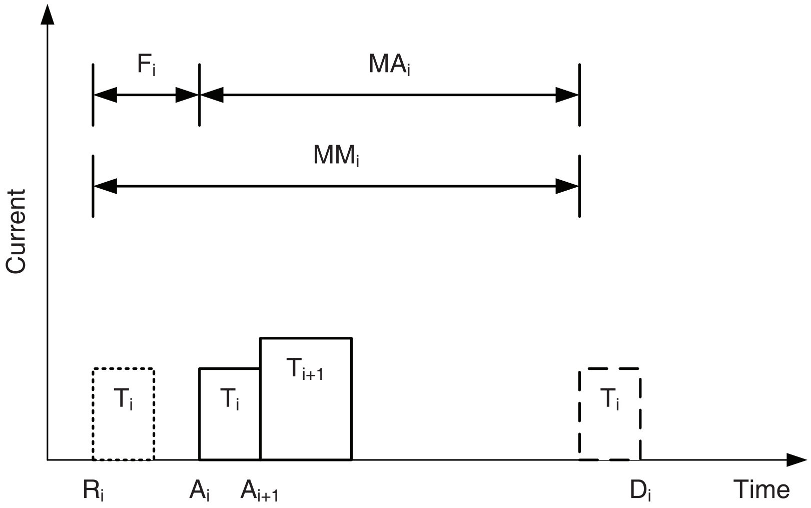


Figure 50: Definition of available delay margin.

$0 < F < MM$ is analyzed to determine the possible delay margin of a task. The cases where $F = 0$ or $F = MM$ can be analyzed in a similar manner. Actually, the results of the $0 < F < MM$ case apply to both cases where $F = 0$ or $F = MM$.

The fact that $0 < F < MM$ does not guarantee that there is some delay margin for a task to exploit. As shown in Figure 50, $0 < F_i < MM_i$ holds for task T_i. However, the end time of T_i is also the ready time of T_{i+1}. Although the execution of T_i can be postponed without missing its deadline, it is possible that the postponed T_{i+1} may miss its deadline. Again, the fundamental rule of adjusting the ready time of a task is to improve the energy performance and not affect the timing performance. It is also a possibility that T_i and T_{i+1} are swapped to take advantage of the delay margin of T_i. This adjustment is not considered in the proposed algorithm. The proposed algorithm sticks to the EDF scheduling rule when the timing constraint is considered. Swapping is against the EDF algorithm and may lead to more deadline misses.

In Figure 50, another timing parameter is defined. The available delay margin is defined as the difference between the maximum delay margin and the ready time flag, as given by Eq. (33). For the two special cases, $MA = MM$ if $F = 0$, and $MA = 0$

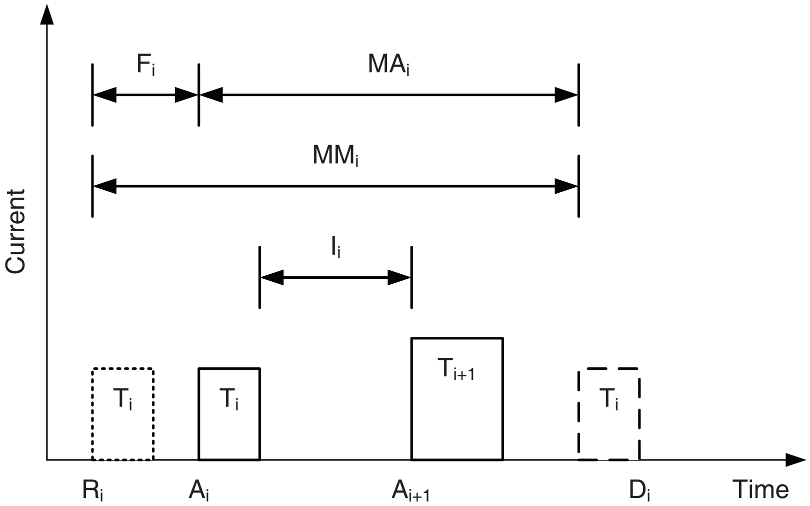


Figure 51: Definition of end time interval.

if $F = MM$.

$$MA = MM - F \tag{33}$$

If a task is not immediately followed by another task, or in other words, the end time of the current task is not equal to the ready time of the next task, there will be some delay margin for the current task. Figure 51 shows one of such scenarios. The end time of T_i is ahead of the ready time of T_{i+1}. A parameter called the end time interval of a task is defined as the difference between its end time and the ready time of the next task, as given by Eq. (34).

$$I_i = A_{i+1} - (A_i + E_i) \tag{34}$$

The end time interval of a task is a measure of the maximum time length by which a task can be delayed for execution before the next task is ready. For the scenario shown in Figure 51, the task T_i can be postponed for execution from its ready time A_i by its end time interval I_i. In this scenario, the available delay margin MA_i is greater than T_i. Although T_i can be executed from the time instant as late as $A_i + MA_i$ to respect its deadline, the maximum delay should be I_i so that the schedule of T_{i+1} is not altered. A parameter called the ready time adjustment margin M is defined to be the minimum value between available delay margin and end time interval, as given

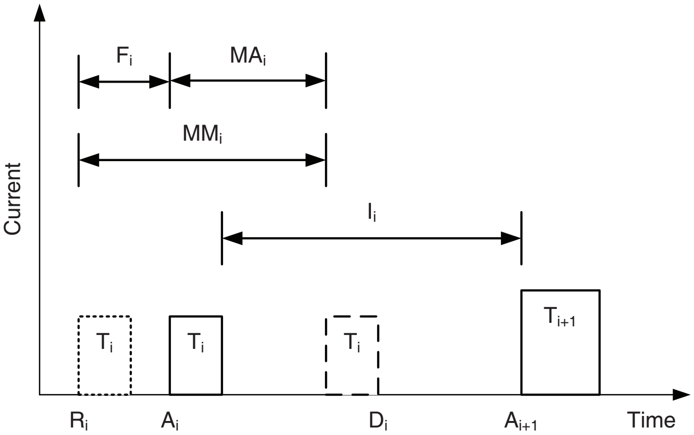


Figure 52: A schedule with available delay margin less than end time interval.

by Eq. (35). For the scenario shown in Figure 50, the end time interval is zero. The definition of ready time adjustment margin also applies.

$$M = min(MA, I) \tag{35}$$

Figure 52 shows the scenario where available delay margin is less than end time interval of a task. For this scenario, ready time adjustment margin is equal to available delay margin. The task T_i must be executed no later than $A_i + MA_i$ to respect its deadline. Therefore, for a task with $0 < F < MM$, the ready time adjustment margin $M = min(MA, I)$. For the special case $F = MM$ and $MA = 0$, the ready time adjustment margin is $M = 0 = min(MA, I)$. For the special case $F = 0$ and $MA = MM$, the ready time adjustment margin is $M = min(MM, I) = min(MA, I)$.

In summary, the ready time adjustment margin of a task is the minimum value between its available delay margin and end time interval when $0 \leq F \leq MM$. For the case $F > MM$, the ready time adjustment margin is always $M = 0$. A task with initial ready time A_i may be executed from any time instant no earlier than A_i and no later than $A_i + M_i$ to respect its deadline. In the meantime, the ready time of the next task is not changed. Algorithm 2 is used to calculate the ready time adjustment margin of a task based on the initial ready time determined using the EDF algorithm shown in Algorithm 1. The ready time adjustment margin is calculated for tasks

$T_1 \leq T_i \leq T_{N-1}$ but not for the last task T_N. This is because the calculation of the end time interval of the current task needs the ready time of the next task, which is not available for the last task. For the last task T_N, the ready time adjustment margin is set to be $M_N = 0$.

Algorithm 2 Calculate Ready Time Adjustment Margin

```
Require: A set of $N$ tasks $T = T_i(R_i, E_i, D_i, W_i)$ and initial schedule with task ready
    time $T_{EDF} = T_i(A_i, E_i, D_i, W_i)$.
for $i = 1 : N - 1$ do
    Ready Time Flag $F_i = A_i - R_i$
    Maximum Delay Margin $MM_i = D_i - R_i - E_i$
    if $F_i > MM_i$ then                          ▷ Deadline is already missed.
        Ready Time Adjustment Margin $M_i = 0$
    else
        Available Delay Margin $MA_i = MM_i - F_i$
        End Time Interval $I_i = A_{i+1} - (A_i + E_i)$
        Ready Time Adjustment Margin $M_i = min(MA_i, I_i)$
    end if
end for
if $i == N$ then
    $M_i = 0$
end if
Algorithm output is Ready Time Adjustment Margin: $M_i$.
```

5.2.3 Determine Ready Time Offset

The actual start time of a task is determined based on the ready time generated using Algorithm 1 and the ready time adjustment margin calculated using Algorithm 2. The actual start time is the sum of the ready time and a new parameter called the ready time offset. The ready time adjustment margin is the length of time by which the start time can be delayed from the ready time. Whether or not this margin is exploited depends on supercapacitor state and energy harvesting. The ready time offset is the actual adjustment made on the ready time. This subsection presents the rules to determine the ready time offset. The rules are based on the results presented in Chapter IV.

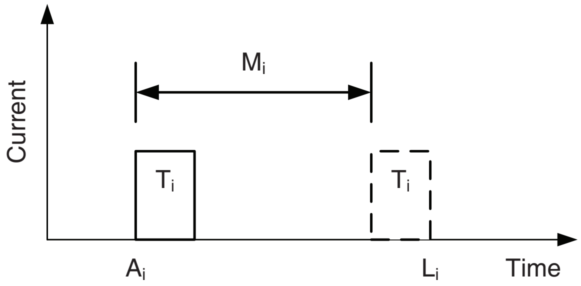


Figure 53: Definition of ready time offset.

Figure 53 shows the ready time and ready time adjustment margin of a task. The start time of this task S_i can be any time instant that satisfies $A_i \leq S_i \leq (A_i + M_i)$. The latest end time of this task L_i is therefore $L_i = A_i + M_i + E_i$. The greedy schedule sets the start time as $S_i = A_i$ and the lazy schedule uses $S_i = A_i + M_i$. Both schedules respect the task deadline. The difference is that one schedule may be preferable if energy constraint is considered. To determine if the greedy or lazy schedule is adopted, supercapacitor initial state at time instant A_i is needed. The initial state is the initial voltages $V_1(t = A_i)$ and $V_2(t = A_i)$. Meanwhile, the energy source profile during the interval between A_i and L_i must be considered.

Based on the results in Chapter IV, the rules to determine which schedule should be used are summarized as follows.

1. **Rule 1:** If the supercapacitor state satisfies the condition that $V_1(t = A) > V_2(t = A)$ and there is no harvested energy during the interval between A and L, the greedy schedule is adopted. The ready time offset is zero: $\Delta = 0$. The start time is the ready time: $S = A$.

2. **Rule 2:** For any other scenarios of supercapacitor state and energy harvesting, the lazy schedule is used. The ready time offset is the ready time adjustment margin: $\Delta = M$. The start time is the ready time plus the ready time offset: $S = A + \Delta$. This start time expression also applies to the first scenario.

The ready time offset is determined using the Algorithm 3. Again, the ready time offset is calculated for tasks $T_1 \leq T_i \leq T_{N-1}$ but not for the last task T_N. For the last task T_N, the ready time offset is set to be $\Delta_N = 0$. Once the ready time offset is known, the start time of a task is calculated as the ready time plus the ready time offset. The initial schedule is modified. The modified schedule maintains the timing performance of the initial schedule and potentially improves the energy performance.

Algorithm 3 Determine Ready Time Offset

```
Require: A set of $N$ tasks: $T = T_i(R_i, E_i, D_i, W_i)$; initial schedule $T_{EDF}$ with ready
    time $A_i$: $T_{EDF} = T_i(A_i, E_i, D_i, W_i)$; ready time adjustment margin: $M_i$; energy
    source model: $E_S(B_S, D_S, W_S)$; and supercapacitor initial state: $V_1(t = 0)$ and
    $V_2(t = 0)$.
Simulation time: $t$
Simulation input 1: Supercapacitor initial state $V_1(t = 0)$ and $V_2(t = 0)$
Simulation input 2: Energy source profile built using $E_S(B_S, D_S, W_S)$
Simulation input 3: Task set profile built using initial schedule $T_{EDF} =$
    $T_i(A_i, E_i, D_i, W_i)$
for $i = 1 : N - 1$ do
    Calculate supercapacitor voltage for $0 \leq t \leq A_i$ using inputs 1-3
    Get supercapacitor state $V_1(t = A_i)$ and $V_2(t = A_i)$
    Latest End Time $L_i = A_i + M_i + E_i$
    if $(V_1(t = A_i) > V_2(t = A_i))\&(W_S(A_i < t < L_i) == 0)$ then          ▷ Rule 1
        Ready Time Offset $\Delta_i = 0$
    else                                                                          ▷ Rule 2
        Ready Time Offset $\Delta_i = M_i$
    end if
    Start Time $S_i = A_i + \Delta_i$
    Modified Schedule $T_{MEDF} = T_i(S_i, E_i, D_i, W_i)$
    Update task set profile using modified schedule $T_{MEDF} = T_i(S_i, E_i, D_i, W_i)$
end for
if $i == N$ then
    Ready Time Offset $\Delta_i = 0$
    Start Time $S_i = A_i$
end if
Algorithm output is modified schedule $T_{MEDF}$ defined by task start time $S_i$:
    $T_{MEDF} = T_i(S_i, E_i, D_i, W_i)$.
```

5.2.4 MEDF Algorithm

The complete MEDF algorithm is summarized in Algorithm 4. The inputs of this algorithm include: a set of N ready but not scheduled tasks: $T = T_i(R_i, E_i, D_i, W_i)$;

energy source model: $E_S(B_S, D_S, W_S)$; and supercapacitor initial state: $V_1(t = 0)$ and $V_2(t = 0)$. The algorithm is a three step process:

1. **Step 1:** Generate an initial schedule using Algorithm 1. The input of this algorithm is the task set $T = T_i(R_i, E_i, D_i, W_i)$. The output is the initial schedule defined by task ready time A_i: $T_{EDF} = T_i(A_i, E_i, D_i, W_i)$.

2. **Step 2:** Calculate ready time adjustment margin of the initial schedule using Algorithm 2. The inputs of Algorithm 2 are the task set $T = T_i(R_i, E_i, D_i, W_i)$ and the initial schedule $T_{EDF} = T_i(A_i, E_i, D_i, W_i)$. The output is the task ready time adjustment margin M_i.

3. **Step 3:** Determine ready time offset of the initial schedule using Algorithm 3. The inputs are: the task set $T = T_i(R_i, E_i, D_i, W_i)$; the initial schedule $T_{EDF} = T_i(A_i, E_i, D_i, W_i)$; ready time adjustment margin M_i; energy source model $E_S(B_S, D_S, W_S)$; and supercapacitor initial state: $V_1(t = 0)$ and $V_2(t = 0)$. The output is the modified schedule T_{MEDF} defined by task start time S_i: $T_{MEDF} = T_i(S_i, E_i, D_i, W_i)$.

5.3 MEDF Algorithm Implementation and Evaluation

5.3.1 Simulation Setup

The MEDF algorithm is implemented based on the system model presented in Section 5.1. The supercapacitor VLR model for the 10 F supercapacitor developed in Chapter II is used as the energy storage model. Its initial state is set to be $V_1(t = 0) = V_2(t = 0) = 1$ V, which is the supercapacitor threshold voltage.

The energy source is modeled as a periodic current pulse train. The period is 100 s. The begin time of each pulse is $B_S(i) = 50 + 100 * (i - 1)$ s. The execution time is same for all the pulses and is set to be $E_S(i) = 10$ s. The weights of the pulses

Algorithm 4 MEDF Algorithm

```
Require: A set of $N$ ready but not scheduled tasks: $T = T_i(R_i, E_i, D_i, W_i)$; energy
    source model: $E_S(B_S, D_S, W_S)$; and supercapacitor initial state: $V_1(t = 0)$ and
    $V_2(t = 0)$.
Step 1: Generate an initial schedule using Algorithm 1.
Input: task set $T$.
Output: initial schedule $T_{EDF}$.
Step 2: Calculate ready time adjustment margin of the initial schedule using
    Algorithm 2.
Input: task set $T$ and initial schedule $T_{EDF}$.
Output: task ready time adjustment margin $M_i$.
Step 3: Determine ready time offset of the initial schedule using Algorithm 3.
Input: task set $T$; initial schedule $T_{EDF}$; ready time adjustment margin $M_i$;
    energy source model: $E_S(B_S, D_S, W_S)$; and supercapacitor initial state: $V_1(t = 0)$
    and $V_2(t = 0)$.
Output: modified schedule $T_{MEDF}$.
MEDF Algorithm Output: modified schedule $T_{MEDF}$ defined by task start time
    $S_i$: $T_{MEDF} = T_i(S_i, E_i, D_i, W_i)$.
```

$W_S(i)$ are uniformly distributed between $[100, 300]$ mA. The current magnitude range is adopted from Section 2.4.4.

Every simulation run creates a task set composed of multiple periodic tasks. A periodic task with two jobs T_1 and T_2 is shown in Figure 54. The release times of T_1 and T_2 are R_1 and R_2, respectively. The interrelease time is defined as the period P of a task: $P = R_2 - R_1$. The execution times of the jobs in a periodic task are equal. Based on the execution time and period of a task, its duty cycle is defined as $DC = E/P$. The release time of the first job in a periodic task is called the phase of a task. For the task shown in Figure 54, the phase $\Phi = R_1$. The release times of the jobs are $R_i = \Phi + P * (i - 1)$. The relative deadline of a job is set to be the period of the task. Therefore, the absolute deadline of each job is $D_i = \Phi + P * i$. For example, the deadline of the first job is $D_1 = \Phi + P$.

In this simulation setup, each task set is composed of five periodic tasks. Each task has five jobs. Therefore, each task set has 25 jobs to be scheduled. The periods are randomly selected from a set {10, 20, 30, . . . , 100} s, each value having an equal

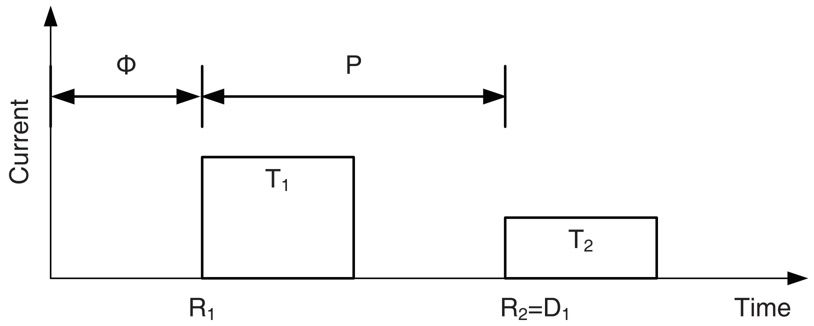


Figure 54: A periodic task with two jobs.

probability of being selected. The duty cycles are selected from a set {0.1, 0.2, 0.3, . . . , 1}, each value having an equal probability of being selected. The phase of a task Φ_i is uniformly distributed between $[0, P_i]$, where P_i is its period. The weights of the jobs are uniformly distributed between $[30, 80]$ mA. This current range is also taken from Section 2.4.4. With a slight abuse of the terminology, the jobs are also called the tasks in this dissertation.

5.3.2 Evaluation Metrics

For each simulation run, two outputs are generated. The task schedule T_{MEDF} created by the MEDF algorithm is used to evaluate the timing performance of this algorithm. The supercapacitor terminal voltage profile V_{MEDF} is used to evaluate the energy performance. As a comparison, the task schedule T_{EDF} and supercapacitor terminal voltage profile V_{EDF} are also generated. The MEDF algorithm is evaluated in terms of two metrics: deadline miss rate α and energy violation rate β.

A task misses its deadline if the completion time is later than its deadline. For the MEDF algorithm, the deadline is missed if $S_i + E_i > D_i$. For the EDF algorithm, the deadline is missed if $A_i + E_i > D_i$. Deadline miss rate α is defined as the ratio between the number of tasks missing their deadlines N_D and the number of tasks to be scheduled N, as given by Eq. (36).

$$\alpha = \frac{N_D}{N} \tag{36}$$

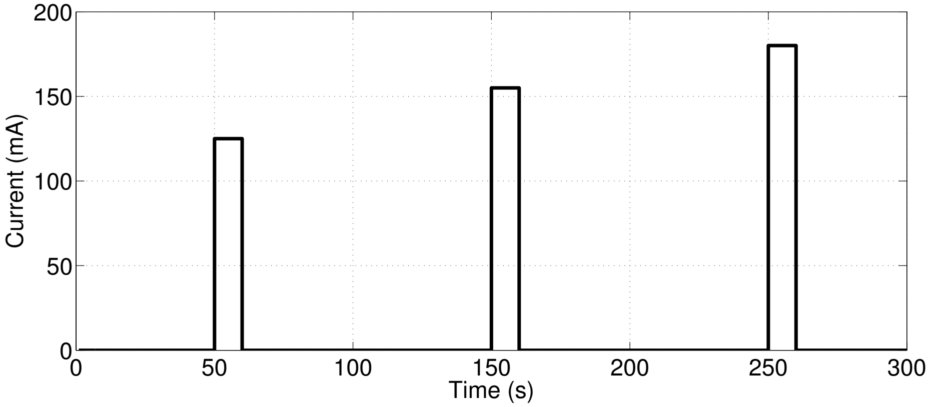


Figure 55: Energy source model for a simulation setup.

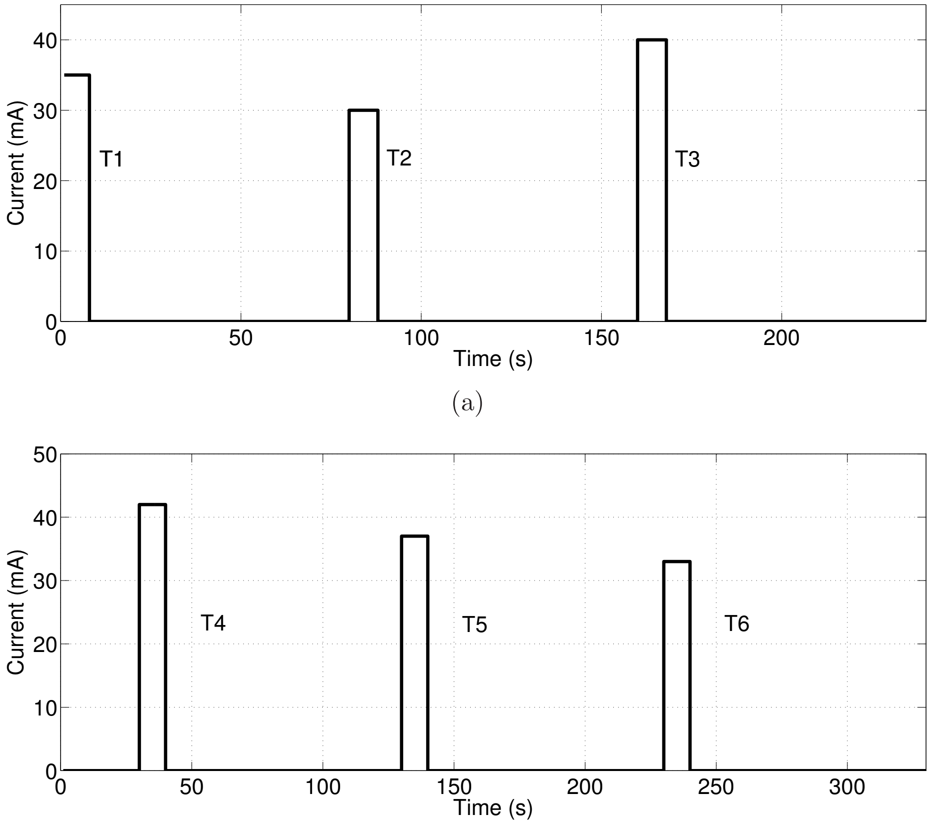
The MEDF algorithm guarantees that the task timing constraints of the EDF schedule are preserved in Step 2. Therefore, it is expected that the timing performance of the MEDF algorithm in terms of deadline miss rate is the same as the EDF algorithm. This observation will be verified using the simulation results.

The execution of a task is considered as an energy violation if the supercapacitor terminal voltage falls below the operation threshold voltage during task execution. For the MEDF algorithm, an energy violation occurs when $min(V_{MEDF}(S_i \leq t \leq (S_i + E_i)) < 1$. For the EDF algorithm, an energy violation occurs if $min(V_{EDF}(A_i \leq t \leq (A_i + E_i)) < 1$. The energy violation rate β is defined as the ratio between the number of tasks having energy violations N_E and the number of tasks to be scheduled N, as given by Eq. (37).

$$\beta = \frac{N_E}{N} \tag{37}$$

5.3.3 An Example

The implementation and evaluation of the MEDF algorithm is illustrated using a simulation setup detailed in this subsection. The supercapacitor initial state is $V_1(t = 0) = V_2(t = 0) = 1$ V. The energy source model is shown in Figure 55 for the first three periods. The period is 100 s. The begin times of the three pulses are 50, 150, and 250 s. The durations are all 10 s. The weights are 125, 155, and 180 mA.



(a)

(b)

Figure 56: Two periodic tasks for a simulation setup. (a) Tasks 1-3. (b) Tasks 4-6.

Figure 56 shows the task set used in this simulation. For simplicity, the task set consists of two periodic tasks and each task is composed of three jobs. All together, six tasks (jobs) need to be scheduled. As shown in Figure 56(a), tasks $T_1 - T_3$ are created using the first periodic task. The period is 80 s, phase is 0 s, duty cycle is 0.1. The execution time is therefore 8 s. Figure 56(b) shows the second periodic task and the last three tasks $T_4 - T_6$. The period is 100 s, phase is 30 s, and execution time is 10 s corresponding to a duty cycle 0.1. The weights of the six tasks are randomly selected between $[30, 80]$ mA. The task parameters are listed in Table 5. The values of the temporal parameters R_i, E_i, and D_i are in the units of seconds. The values of weights W_i are in the units of mA.

The EDF schedule determined using Algorithm 1 is shown in Figure 57. All the

Table 5: Tasks to Be Scheduled

	T_1	T_2	T_3	T_4	T_5	T_6
Release Time (R_i)	0	80	160	30	130	230
Execution Time (E_i)	8	8	8	10	10	10
Deadline (D_i)	80	160	240	130	230	330
Weight (W_i)	35	30	40	42	37	33

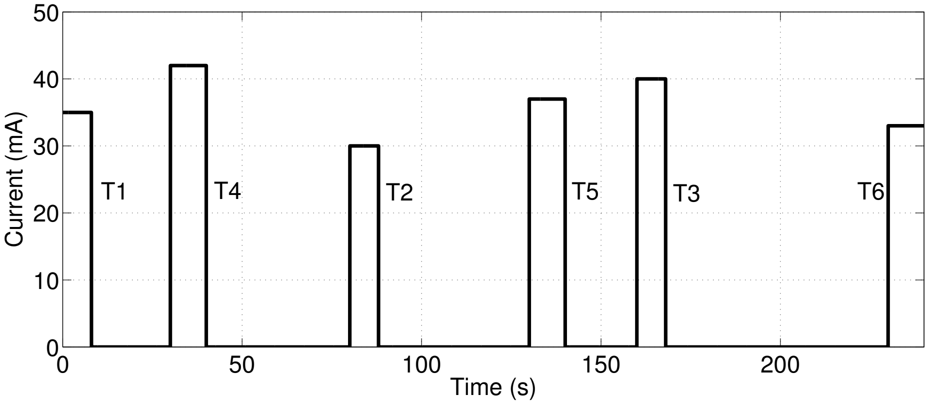


Figure 57: Task schedule determined using EDF algorithm.

tasks are executed immediately after it is released. All the deadlines are respected. Therefore, the deadline miss rate is $\alpha_{EDF} = 0$. The supercapacitor terminal voltage profile is shown in Figure 58. Three energy violations occur: T_1, T_4, and T_5. The minimum supercapacitor terminal voltages during the executions of the three tasks are 0.9670 V, 0.9216 V, and 0.9888 V, respectively. The energy violation rate is therefore $\beta_{EDF} = 3/6 = 0.5$.

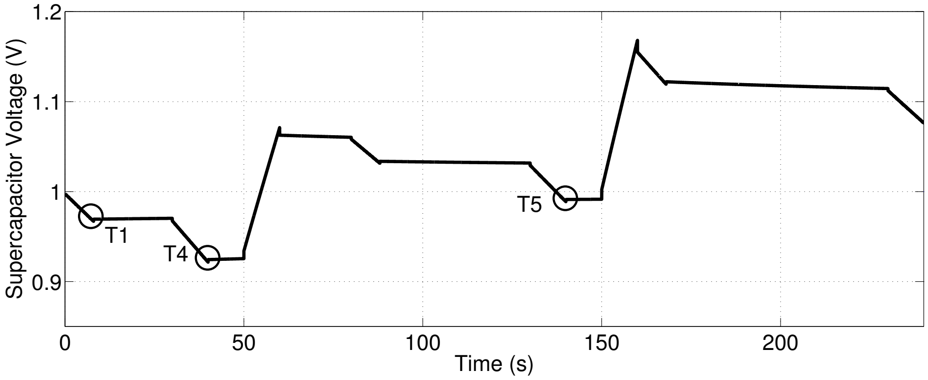


Figure 58: Supercapacitor terminal voltage profile of EDF schedule.

Table 6: EDF Schedule and Ready Time Adjustment Margin

EDF Schedule	T_1	T_4	T_2	T_5	T_3	T_6
EDF Schedule Ready Time (A_i)	0	30	80	130	160	230
Ready Time Flag ($F_i = A_i - R_i$)	0	0	0	0	0	0
Maximum Delay Margin ($MM_i = D_i - R_i - E_i$)	72	90	72	90	72	90
Available Delay Margin ($MA_i = MM_i - F_i$)	72	90	72	90	72	90
End Time Interval ($I_i = A_{i+1} - (A_i + E_i)$)	22	40	42	20	62	N/A
Ready Time Adjustment Margin $M_i = min(MA_i, I_i)$	22	40	42	20	62	N/A

Algorithm 2 is executed to determine the task ready time adjustment margin based on the EDF schedule. Table 6 lists the calculated parameters. Algorithm 3 is run to calculate the task ready time offset. The results are shown in Table 7. The task start time is then determined and the MEDF schedule is finalized. The MEDF schedule is shown in Figure 59. Tasks T_1, T_4, and T_5 are postponed for execution. Although some tasks are delayed for execution, all the deadlines are still respected. Therefore, the deadline miss rate is again $\alpha_{MEDF} = 0$. The MEDF and EDF algorithms have the same timing performance. The supercapacitor terminal voltage profile is shown in Figure 60. One energy violation occurs. The minimum supercapacitor voltage during the T_1 execution is 0.9670 V, which results in an unsuccessful execution. The energy violation rate is $\beta_{MEDF} = 1/6 = 0.167$. This example demonstrates that the MEDF algorithm is better in terms of energy performance than the EDF algorithm while maintaining the same timing performance.

5.3.4 Evaluation Results

The simulations are run for 200 times using the setup specified in Section 5.3.1. The deadline miss rates and energy violation rates are recorded for the MEDF and EDF schedules. The obtained evaluation metrics are sorted in the ascending orders and plotted. As shown in Figure 61, 120 out of the 200 simulation runs have zero deadline miss rates. The EDF and MEDF algorithms always have the same deadline

Table 7: MEDF Schedule and Ready Time Offset

EDF Schedule	T_1	T_4	T_2	T_5	T_3	T_6
EDF Schedule Ready Time (A_i)	0	30	80	130	160	230
Ready Time Adjustment Margin $M_i = min(MA_i, I_i)$	22	40	42	20	62	N/A
Execution Time (E_i)	8	10	8	10	8	10
Latest End Time $L_i = A_i + M_i + E_i$	30	80	130	160	230	N/A
$V_1(t = A_i)$	1	0.9693	1.0575	0.8947	1.1554	N/A
$V_2(t = A_i)$	1	0.9988	1.0130	0.9974	1.0277	N/A
If $V_1(t = A_i) > V_2(t = A_i)$	False	False	True	False	True	N/A
If $W_S(A_i < t < L_i) == 0$	True	False	True	False	True	N/A
Ready Time Offset (Δ_i)	22	40	0	20	0	N/A
MEDF Start Time ($S_i = A_i + \Delta_i$)	22	70	80	150	160	230

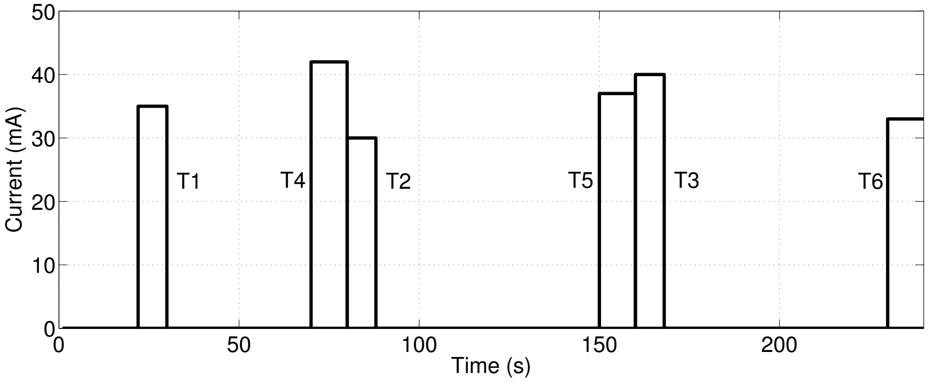


Figure 59: Task schedule determined using MEDF algorithm.

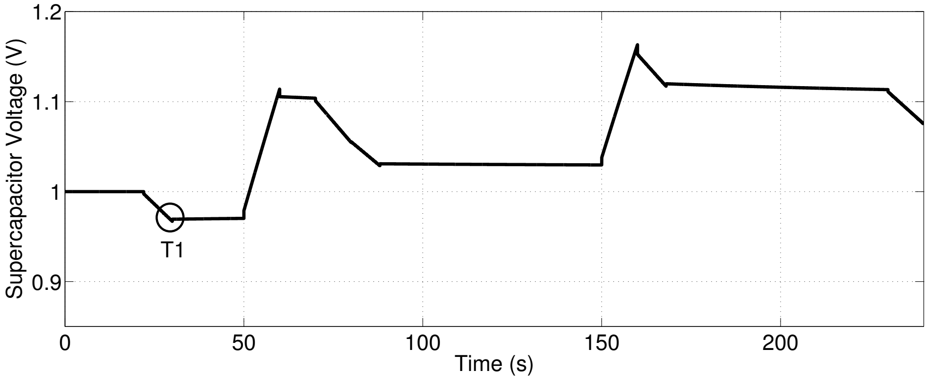


Figure 60: Supercapacitor terminal voltage profile of MEDF schedule.

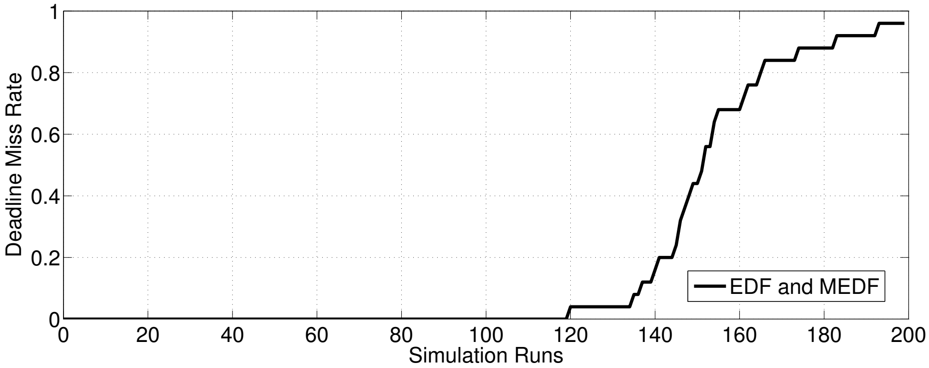


Figure 61: Deadline miss rates of EDF and MEDF algorithms.

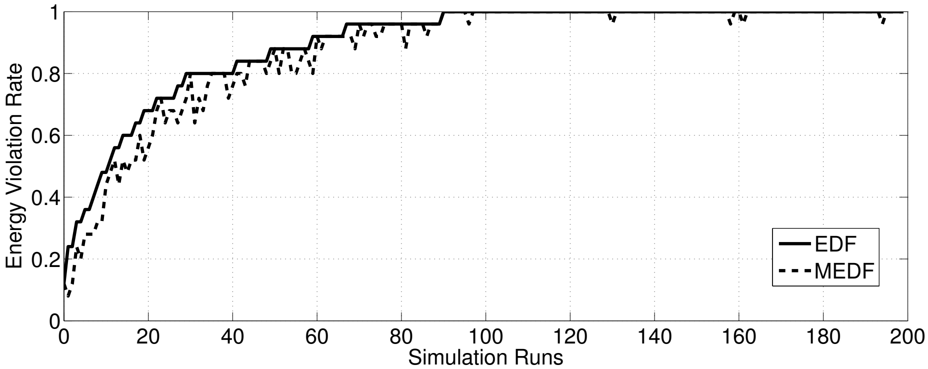


Figure 62: Energy violation rates of EDF and MEDF algorithms.

miss rates. This is due to the rules used to develop the MEDF algorithm. The task timing constraints of the EDF algorithm are preserved in the MEDF algorithm. The energy violation rates are shown in Figure 62. For the EDF algorithm, 110 out of the 200 simulation runs have an energy violation rate $\beta_{EDF} = 1$. Among the 110 runs, the MEDF schedules have an energy violation rate less than one for five runs. Among the other 90 runs, the MEDF algorithm results in an energy violation rate smaller than that of the EDF algorithm for 54 runs. All together, the MEDF schedules result in a smaller energy violation rate for 59 runs and a same energy violation rate for 141 runs. The simulation results again verify that the MEDF algorithm improves the energy performance of the EDF algorithm and maintains the timing performance at the same time.

To quantitatively evaluate the improvement of the MEDF algorithm over the EDF algorithm in terms of energy violation rate, the simulation setup is slightly modified. As defined in [38, 44], the utilization of a scheduler is

$$U = \sum_{i=1}^{N} \frac{E_i}{P_i}, \tag{38}$$

where N is the number of the periodic tasks, E_i and P_i are the execution times and periods, respectively. In this simulation setup, $N = 5$. For each task set, the duty cycles of the five periodic tasks are set to be equal. Therefore, the utilization is $U = 5 * DC$, where DC is the duty cycle of a periodic task. The duty cycle varies from 0.02 to 0.14 with a step of 0.02. The utilization is therefore scanned from 0.1 to 0.7 with a step of 0.1. The periods, phases, and weights of the tasks are defined using the same setup specified in Section 5.3.1.

The mean absolute percentage error (MAPE) is used to characterize the improvement of the MEDF algorithm over the EDF algorithm in terms of energy violation rate. The MAPE is defined as follows:

$$MAPE = \frac{1}{M} \sum_{j=1}^{M} \frac{|\beta_{MEDF}(j) - \beta_{EDF}(j)|}{\beta_{EDF}(j)} * 100\%, \tag{39}$$

where M is the number of simulation runs. The MAPE is a measure of the extent to which the energy violation rate of the EDF algorithm can be reduced if the MEDF algorithm is used. In this simulation setup, 30 simulations are run for each utilization and therefore $M = 30$.

The calculated MAPE values for different utilizations are plotted in Figure 63. The MAPE increases when the utilization decreases. The average MAPE is 17.5% for the utilizations shown in Figure 63, which means that the average energy violation rate of the EDF algorithm is reduced by 17.5%. The MAPE is 37% for utilization $U = 0.1$ and decreases all the way to 0.8% for $U = 0.7$. Therefore, the MEDF algorithm improves the EDF algorithm in terms of energy violation rate more significantly when a sensor node is lightly loaded. Wireless sensor nodes usually operate periodically.

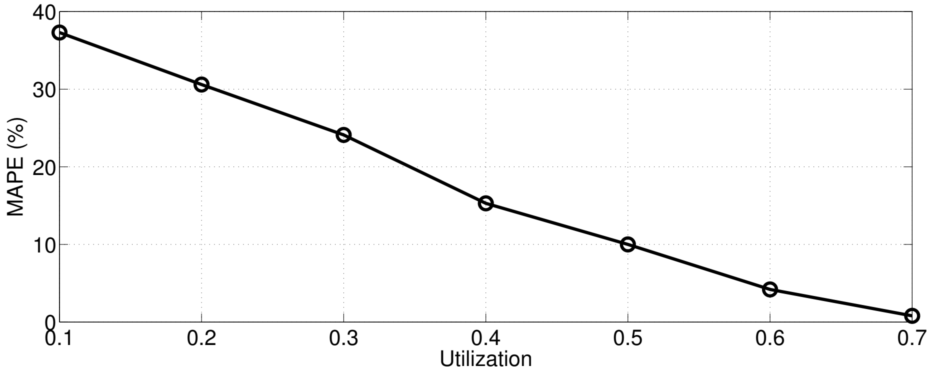


Figure 63: MAPE versus utilization for MEDF algorithm.

If the duty cycle of a sensor node is relatively low, the MEDF algorithm can better utilize the harvested energy than the EDF algorithm.

5.4 Summary

This chapter presents the MEDF algorithm for independent tasks, which are non-preemptable tasks without precedence constraints. Section 5.1 introduces the system model for a supercapacitor based environmentally powered wireless sensor node. The system model has three components: energy source, energy storage, and energy consumer. Section 5.2 presents the MEDF algorithm. The MEDF algorithm has three steps. First, generate an initial schedule using the EDF algorithm. This step takes care of the timing constraints of tasks. Second, calculate task ready time adjustment margin based on the initial schedule. This margin determines how much delay is allowed if task ready time of the initial schedule is adjusted. Third, task ready time offset is determined based on supercapacitor state and energy harvesting. Task start time is task ready time plus the ready time offset. The MEDF algorithm takes into account the energy constraints of tasks in addition to timing constraints. In Section 5.3, the MEDF algorithm is implemented and evaluated in terms of two metrics: deadline miss rate and energy violation rate. Simulation results show that the MEDF algorithm improves the energy performance of the EDF algorithm while maintaining the same timing performance.

CHAPTER VI

SCHEDULING TASKS WITH PRECEDENCE CONSTRAINTS

This chapter continues the quantitative study of the task scheduling problem in supercapacitor based environmentally powered wireless sensor nodes. Nonpreemptable tasks with precedence constraints are considered. Some tasks in a wireless sensor node may be constrained to execute in certain order. For example, a sensor node can only transmit data packets after sensing the environment and processing the data.

A task scheduling algorithm for nonpreemptable tasks with precedence constraints is presented in this chapter. Section 6.1 proposes the modified first in first out (MFIFO) algorithm. The MFIFO algorithm takes into account supercapacitor state and energy harvesting. Task precedence constraints are handled by defining a variable called task effective release time. The MFIFO algorithm is also a three step process. First, the first in first out (FIFO) algorithm is used to create an initial schedule. The second and third steps are similar to those of the MEDF algorithm. Section 6.2 implements and evaluates the MFIFO algorithm. Results show that the MFIFO algorithm improves the energy performance of the FIFO algorithm and maintains the timing performance at the same time. Section 6.3 is a summary.

6.1 Modified First In First Out (MFIFO) Algorithm

The system model presented in Section 5.1 is used in this chapter to develop the MFIFO algorithm. The energy source and energy storage models are exactly the same as the ones defined in Section 5.1. The energy consumer model is modified to take into account task precedence constraints.

6.1.1 Task Precedence Constraint and Effective Release Time

In addition to the four parameters (release time, execution time, deadline, and weight) used to characterize the task model presented in Section 5.1, a task may also have precedence constraints. If tasks are constrained to execute in some order, they are said to have precedence constraints. The precedence constraints among tasks are specified using precedence relations [37]. A task T_p is a predecessor of another task T_q (and T_q a successor of T_p) if T_q cannot begin execution until the execution of T_p completes. This fact is usually denoted by $T_p < T_q$. Two tasks are independent when neither $T_p < T_q$ or $T_p > T_q$. A task with predecessors is ready for execution when the time is at or after its release time and executions of all the predecessors are completed. Without loss of generality, it is assumed that in this chapter a task may have no more than one predecessor or successor for simplicity.

The release times of tasks with precedence constraints are sometimes inconsistent with the precedence constraints, which means that the release time of a task may be later than that of its successor. Figure 64 shows two tasks T_p and T_q using solid lines. If $T_p < T_q$, the release time of task T_q is earlier than the release time of T_p, which is not consistent with the precedence constraint. A parameter called the effective release time of a task is defined to deal with such inconsistency. The effective release time of a task without predecessor is equal to its release time. The effective release time of a task with predecessor is equal to the maximum value between its release time and the release time of its predecessor plus the execution time of its predecessor. For example, the effective release time ER_q of T_q is defined by Eq. (40) depending on if there is a precedence constraint. As shown in Figure 64, the task T_q denoted by dashed lines shows its effective release time if $T_p < T_q$.

$$ER_q = \begin{cases} max(R_q, R_p + E_p), & T_p < T_q; \\ R_q, & otherwise. \end{cases} \tag{40}$$

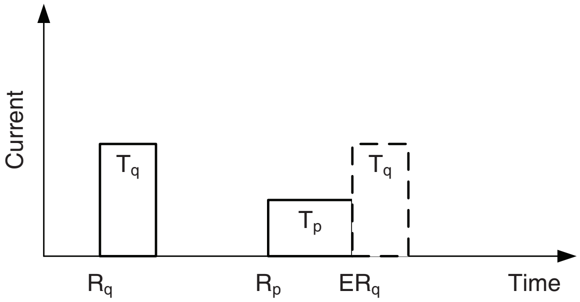


Figure 64: Definition of task effective release time.

6.1.2 Create An Initial Schedule Using FIFO Algorithm

The FIFO algorithm is used to create an initial schedule instead of the EDF algorithm used in Chapter V to ensure that the task precedence constraints are satisfied. The tasks are originally defined by the task set $T = T_i(R_i, E_i, D_i, W_i)$ and the precedence constraints $T_p < T_q$. The precedence constraints are transformed into timing constraints by defining the effective release times. A task is then characterized by four parameters: effective release time ER, execution time E, deadline D, and weight W. The task set is now $T_E = T_i(ER_i, E_i, D_i, W_i)$. The FIFO algorithm sorts the effective release times in the ascending order and determines the ready times of the tasks. The initial schedule is determined using Algorithm 5.

6.1.3 MFIFO Algorithm

Once the initial schedule is determined, the ready time adjustment margin and ready time offset can be calculated using Algorithms 2 and 3 shown in Chapter V, respectively. It should be noted that the release times used in Algorithms 2 and 3 should be replaced by the effective release times. The complete MFIFO algorithm is summarized in Algorithm 6. The inputs of this algorithm include: a set of N ready but not scheduled tasks: $T = T_i(R_i, E_i, D_i, W_i)$; task precedence constraints: $T_p < T_q$; energy source model: $E_S(B_S, D_S, W_S)$; and supercapacitor initial state: $V_1(t = 0)$ and $V_2(t = 0)$. The MFIFO algorithm is a three step process:

Algorithm 5 Create An Initial Schedule Using FIFO Algorithm

```
Require: A set of $N$ ready but not scheduled tasks: $T = T_i(R_i, E_i, D_i, W_i)$ and task
    precedence constraints: $T_p < T_q$.
for $i = 1 : N$ do
    if $T_p < T_q$ then
        $ER_q = max(R_q, R_p + E_p)$
    else
        $ER_q = R_q$
    end if
end for
Sort $N$ tasks in the ascending order of their effective release times.
Current Time $T_C = 0$
for $i = 1 : N$ do
    Ready Time $A_i = max(T_C, ER_i)$
    Current Time $T_C = A_i + E_i$
end for
Algorithm output is initial schedule $T_{FIFO}$ defined by task ready time $A_i$: $T_{FIFO} =$
    $T_i(A_i, E_i, D_i, W_i)$ and modified task set $T_E = T_i(ER_i, E_i, D_i, W_i)$.
```

1. **Step 1:** Create an initial schedule using Algorithm 5. The input of this algorithm is the task set $T = T_i(R_i, E_i, D_i, W_i)$ and task precedence constraints $T_p < T_q$. The output is the initial schedule T_{FIFO} defined by task ready time A_i: $T_{FIFO} = T_i(A_i, E_i, D_i, W_i)$ and modified task set $T_E = T_i(ER_i, E_i, D_i, W_i)$.

2. **Step 2:** Calculate ready time adjustment margin of the initial schedule using Algorithm 2. The inputs of Algorithm 2 are the modified task set $T_E = T_i(ER_i, E_i, D_i, W_i)$ and the initial schedule $T_{FIFO} = T_i(A_i, E_i, D_i, W_i)$. The output is the task ready time adjustment margin M_i.

3. **Step 3:** Determine ready time offset of the initial schedule using Algorithm 3. The inputs are: the modified task set $T_E = T_i(ER_i, E_i, D_i, W_i)$; the initial schedule $T_{FIFO} = T_i(A_i, E_i, D_i, W_i)$; the ready time adjustment margin M_i; the energy source model $E_S(B_S, D_S, W_S)$; and the supercapacitor initial state $V_1(t = 0)$ and $V_2(t = 0)$. The output is the modified schedule T_{MFIFO} defined by task start time S_i: $T_{MFIFO} = T_i(S_i, E_i, D_i, W_i)$.

Algorithm 6 MFIFO Algorithm

```
Require: A set of $N$ ready but not scheduled tasks: $T = T_i(R_i, E_i, D_i, W_i)$; task precedence constraints: $T_p < T_q$; energy source model: $E_S(B_S, D_S, W_S)$; and supercapacitor initial state: $V_1(t = 0)$ and $V_2(t = 0)$.
Step 1: Create an initial schedule using Algorithm 5.
Input: task set $T$ and task precedence constraints $T_p < T_q$.
Output: initial schedule $T_{FIFO}$ and modified task set $T_E$.
Step 2: Calculate ready time adjustment margin of the initial schedule using Algorithm 2.
Input: modified task set $T_E$ and initial schedule $T_{FIFO}$.
Output: task ready time adjustment margin $M_i$.
Step 3: Determine ready time offset of the initial schedule using Algorithm 3.
Input: modified task set $T_E$; initial schedule $T_{FIFO}$; ready time adjustment margin $M_i$; energy source model: $E_S(B_S, D_S, W_S)$; and supercapacitor initial state: $V_1(t = 0)$ and $V_2(t = 0)$.
Output: modified schedule $T_{MFIFO}$.
MFIFO Algorithm Output: modified schedule $T_{MFIFO}$ defined by task start time $S_i$: $T_{MFIFO} = T_i(S_i, E_i, D_i, W_i)$.
```

6.2 MFIFO Algorithm Implementation and Evaluation

6.2.1 Simulation Setup

The MFIFO algorithm is implemented and evaluated using a simulation setup similar to the one used for the MEDF algorithm presented in Section 5.3.1. The energy source and energy storage models are exactly the same. The energy consumer model is modified. Each task set has six periodic tasks and each periodic task has five jobs. Therefore, each task set is composed of 30 tasks. The timing and energy parameters of a task are defined in the same way as the one used for the MEDF algorithm, too. The precedence constraints are assigned with controlled randomness. The six periodic tasks are grouped into three groups. Each group consists of two periodic tasks. For convenience, the six periodic tasks are numbered as $\{P_1, P_2, \ldots, P_6\}$. The three groups are then $\{P_1, P_2\}$, $\{P_3, P_4\}$, and $\{P_5, P_6\}$. For each group, a job of the first periodic task is randomly selected as the predecessor of a job randomly selected from the second periodic task. Therefore, three pairs of precedence constraints are assigned for each task set. For example, Figure 65 shows that in the first group

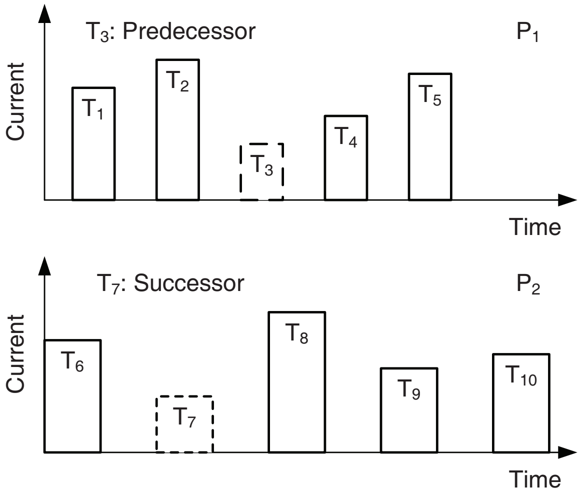


Figure 65: Assignment of precedence constraint.

$\{P_1, P_2\}$, the third job T_3 of the first periodic task P_1 is selected as the predecessor of the second job T_7 of the second periodic task P_2: $T_3 < T_7$. The MFIFO algorithm is also evaluated in terms of the two metrics used for the MEDF algorithm: deadline miss rate and energy violation rate.

6.2.2 An Example

An example is used to illustrate the implementation and evaluation of the MFIFO algorithm. The simulation setup is adopted from Section 5.3.3, which is used to illustrate the MEDF algorithm implementation and evaluation. The energy source model uses the model shown in Figure 55. The supercapacitor initial state is $V_1(t = 0) = V_2(t = 0) = 1$ V. The two periodic tasks shown in Figure 56 are used to define the task timing and energy constraints. The job T_2 from the first periodic task is selected as the predecessor of the job T_4 from the second periodic task. The precedence constraint is therefore $T_2 < T_4$. The effective release time of task T_4 is $ER_4 = max(R_4, R_2 + E_2) = 88$ s. The effective release times of the other five tasks are their release times. The task characteristics are listed in Table 8.

The FIFO schedule determined using Algorithm 5 is shown in Figure 66. All the tasks are scheduled for execution at their effective release times. The task T_4 begins

Table 8: Tasks With Precedence Constraints to Be Scheduled

	T_1	T_2	T_3	T_4	T_5	T_6
Release Time (R_i)	0	80	160	30	130	230
Effective Release Time (ER_i)	0	80	160	88	130	230
Execution Time (E_i)	8	8	8	10	10	10
Deadline (D_i)	80	160	240	130	230	330
Weight (W_i)	35	30	40	42	37	33

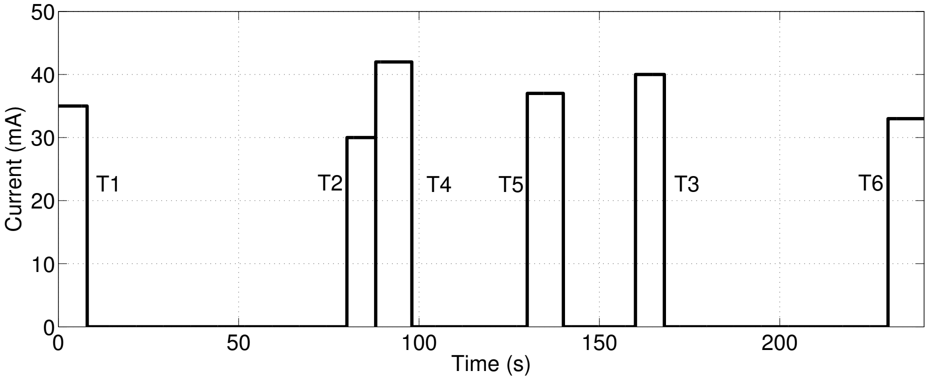


Figure 66: Task schedule determined using FIFO algorithm.

execution when its predecessor T_2 completes execution. The precedence constraint is satisfied. All the tasks respect their deadlines. The deadline miss rate is therefore $\alpha_{FIFO} = 0$. The supercapacitor terminal voltage profile is shown in Figure 67. Two energy violations occur: T_1 and T_5. The minimum supercapacitor terminal voltages during the executions of the two tasks are 0.9670 V and 0.9867 V, respectively. The energy violation rate is therefore $\beta_{FIFO} = 2/6 = 0.333$.

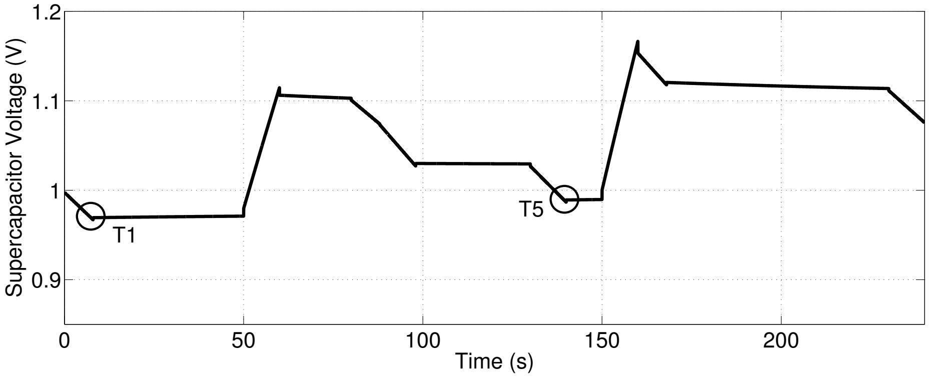


Figure 67: Supercapacitor terminal voltage profile of FIFO schedule.

Table 9: FIFO Schedule and Ready Time Adjustment Margin

FIFO Schedule	T_1	T_2	T_4	T_5	T_3	T_6
FIFO Schedule Ready Time (A_i)	0	80	88	130	160	230
Ready Time Flag ($F_i = A_i - ER_i$)	0	0	0	0	0	0
Maximum Delay Margin $MM_i = D_i - ER_i - E_i$	72	72	32	90	72	90
Available Delay Margin ($MA_i = MM_i - F_i$)	72	72	32	90	72	90
End Time Interval ($I_i = A_{i+1} - (A_i + E_i)$)	72	0	32	20	62	N/A
Ready Time Adjustment Margin $M_i = min(MA_i, I_i)$	72	0	32	20	62	N/A

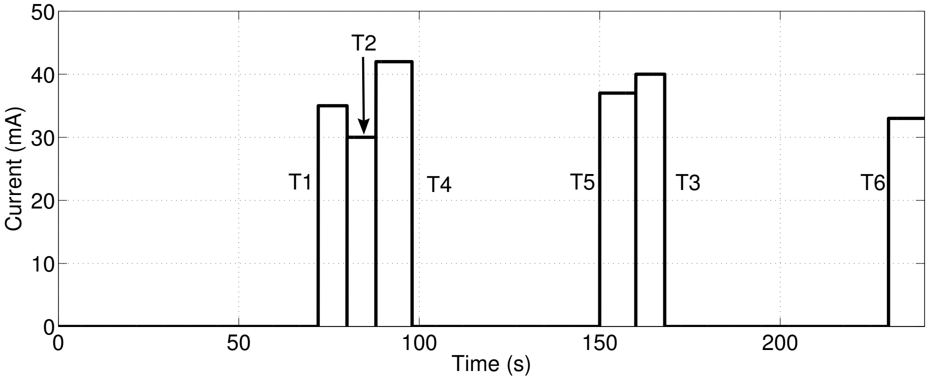


Figure 68: Task schedule determined using MFIFO algorithm.

The task ready time adjustment margin based on the FIFO schedule is calculated using Algorithm 2. Table 9 lists the calculated parameters. Algorithm 3 is run to calculate the task ready time offset. The results are shown in Table 10. The task start time is then determined and the MFIFO schedule is finalized. The MFIFO schedule is shown in Figure 68. Tasks T_1 and T_5 are postponed for execution. All the deadlines are respected. Therefore, the deadline miss rate is still $\alpha_{MFIFO} = 0$. The MFIFO and FIFO algorithms have the same timing performance. The supercapacitor terminal voltage profile is shown in Figure 69. No energy violation occurs. The energy violation rate is $\beta_{MFIFO} = 0$. This example demonstrates that the MFIFO algorithm is better than the FIFO algorithm in terms of energy performance while maintaining the same timing performance.

Table 10: MFIFO Schedule and Ready Time Offset

FIFO Schedule	T_1	T_2	T_4	T_5	T_3	T_6
FIFO Schedule Ready Time (A_i)	0	80	88	130	160	230
Ready Time Adjustment Margin $M_i = min(MA_i, I_i)$	72	0	32	20	62	N/A
Execution Time (E_i)	8	8	10	10	8	10
Latest End Time $L_i = A_i + M_i + E_i$	80	88	130	160	230	N/A
$V_1(t = A_i)$	1	1.1005	1.0738	0.8825	1.1539	N/A
$V_2(t = A_i)$	1	1.0247	1.0287	1.0109	1.0352	N/A
If $V_1(t = A_i) > V_2(t = A_i)$	False	True	True	False	True	N/A
If $W_S(A_i < t < L_i) == 0$	False	True	True	False	True	N/A
Ready Time Offset (Δ_i)	72	0	0	20	0	N/A
MFIFO Start Time $S_i = A_i + \Delta_i$	72	80	88	150	160	230

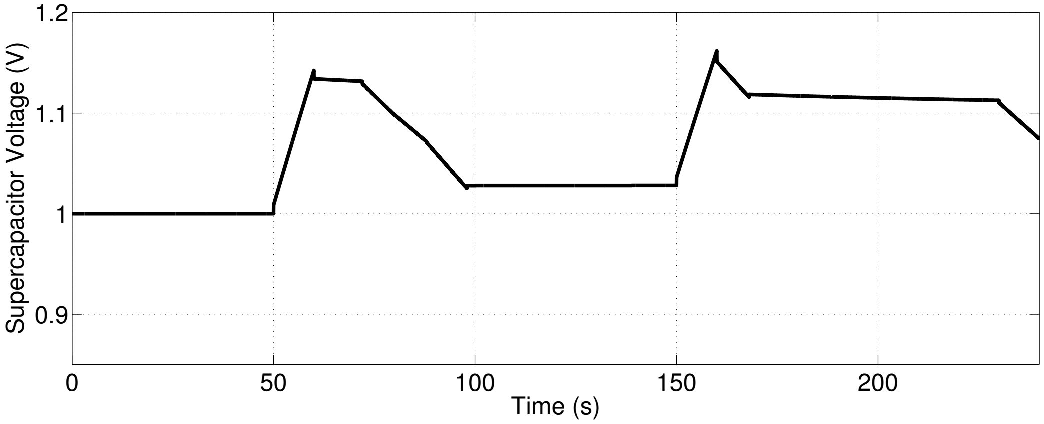


Figure 69: Supercapacitor terminal voltage profile of MFIFO schedule.

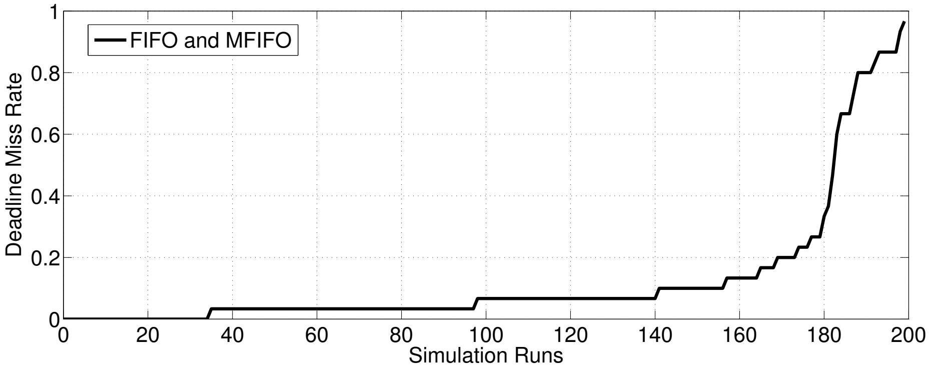


Figure 70: Deadline miss rates of FIFO and MFIFO algorithms.

6.2.3 Evaluation Results

The simulations are run for 200 times using the setup specified in Section 6.2.1. The deadline miss rates and energy violation rates are recorded for the FIFO and MFIFO schedules. The obtained evaluation metrics are sorted in the ascending orders and plotted. As shown in Figure 70, 35 out of the 200 simulation runs have zero deadline miss rates. The FIFO and MFIFO algorithms always have the same deadline miss rates. The timing and precedence constraints of the FIFO schedules are preserved in the MFIFO schedules. The energy violation rates are shown in Figure 71. For the FIFO algorithm, 96 out of the 200 simulation runs have an energy violation rate $\beta_{FIFO} = 1$. Among the 96 runs, the MFIFO schedules have an energy violation rate less than one for seven runs. Among the other 104 runs, the MFIFO algorithm results in an energy violation rate smaller than that of the FIFO algorithm for 81 runs. All together, the MFIFO schedules result in a smaller energy violation rate for 88 runs and a same energy violation rate for 112 runs. The simulation results verify that the MFIFO algorithm improves the energy performance of the FIFO algorithm and maintains the timing performance at the same time.

The simulation setup is slightly modified to quantitatively compare the energy violation rates of the FIFO and MFIFO algorithms. In Eq. (38), $N = 6$ for this simulation setup. The duty cycles of the six periodic tasks take the same value for

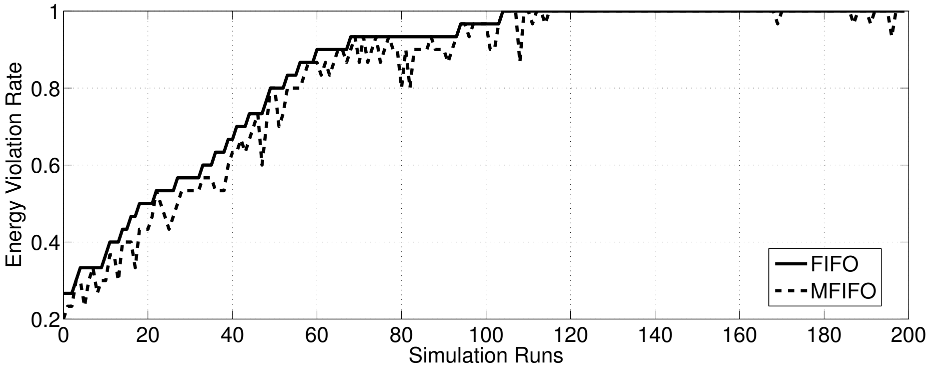


Figure 71: Energy violation rates of FIFO and MFIFO algorithms.

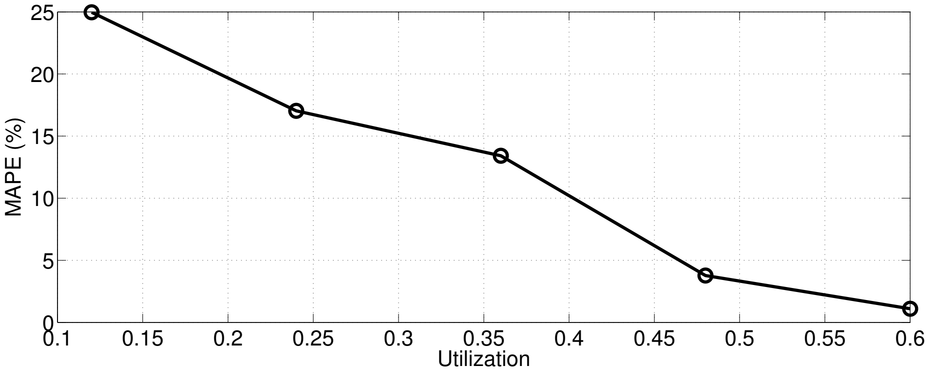


Figure 72: MAPE versus utilization for MFIFO algorithm.

each utilization. The utilization is $U = 6 * DC$. The duty cycle increases from 0.02 to 0.1 with a step of 0.02. The utilization is scanned from 0.12 to 0.6 with a step of 0.12. The other parameters of the tasks including periods, phases, weights, and precedence constraints are still defined using the setup specified in Section 6.2.1. The simulations are run for 30 times for each utilization. In Eq. (39), $M = 30$. Figure 72 shows the calculated MAPE values for the different utilizations. The MAPE decreases as the utilization increases. The average MAPE for the five utilizations is 12.1%. The MFIFO algorithm reduces the average energy violation rate of the FIFO algorithm by 12.1%. The MAPE is 25% for utilization $U = 0.12$. Like the MEDF algorithm, the MFIFO algorithm improves the energy violation rate more significantly if the sensor node operates with a relatively low duty cycle.

6.3 Summary

This chapter proposes the MFIFO algorithm for nonpreemptable tasks with precedence constraints. Section 6.1 presents the MFIFO algorithm. The task precedence constraints are transformed into timing constraints by defining the effective release time of a task. The effective release time of a task with predecessor is equal to the maximum value between its release time and the release time of its predecessor plus the execution time of its predecessor. The MFIFO algorithm has three steps. First, create an initial schedule using the FIFO algorithm. This step takes care of the timing constraints and precedence constraints of tasks. Second, calculate the ready time adjustment margin based on the initial schedule. This margin determines how much delay is allowed if the ready time of the initial schedule is adjusted. Third, the ready time offset is determined based on supercapacitor state and energy harvesting. The start time of a task is the ready time plus the ready time offset. The MFIFO algorithm takes into account the energy constraints of tasks in addition to the timing constraints. The MFIFO algorithm is implemented and evaluated in Section 6.2. An example is shown to illustrate how the MFIFO algorithm works. Simulation results show that the MFIFO algorithm improves the energy performance of the FIFO algorithm and maintains the timing performance at the same time.

CHAPTER VII

CONCLUSION

7.1 Dissertation Contributions

This dissertation considers the task scheduling problem in wireless sensor nodes that harvest energy from ambient environment and use supercapacitor based storage systems to buffer the harvested energy. This dissertation makes five contributions. First, a physics based equivalent circuit model for supercapacitors is developed. The variable leakage resistance (VLR) model takes into account three mechanisms of supercapacitors: voltage dependency of capacitance, charge redistribution, and self-discharge. Second, the effects of time and supercapacitor initial state on supercapacitor voltage change and energy loss during charge redistribution are investigated. Third, the task scheduling problem in supercapacitor based environmentally powered wireless sensor nodes is studied qualitatively. The impacts of supercapacitor state and energy harvesting on task scheduling are examined. Task scheduling rules are developed. Fourth, the task scheduling problem in supercapacitor based environmentally powered wireless sensor nodes is studied quantitatively. The modified earliest deadline first (MEDF) algorithm is developed to schedule nonpreemptable tasks without precedence constraints. Finally, the modified first in first out (MFIFO) algorithm is proposed to schedule nonpreemptable tasks with precedence constraints. The MEDF and MFIFO algorithms take into account energy constraints of tasks in addition to timing constraints. The MEDF and MFIFO algorithms improve the energy performance and maintain the timing performance of the earliest deadline first (EDF) and first in first out (FIFO) algorithms, respectively. The contributions are summarized as follows.

Chapter II presents the VLR model for supercapacitors. The VLR model is an equivalent circuit model that takes into account three mechanisms of supercapacitors: voltage dependency of capacitance, charge redistribution, and self-discharge. The characterization procedures for the VLR model parameters are illustrated using a 10 F sample supercapacitor. A MATLAB Simulink model is implemented to determine the variable leakage resistance using the measured supercapacitor voltage during a self-discharge experiment. The variable leakage resistance is related to the supercapacitor terminal voltage by a piecewise linear function. Four supercapacitor models (VLR, EIE, two branch model, and three branch model) are evaluated by comparing the simulated supercapacitor voltages using these models and the measured voltage during three experiments: charging-redistribution, self-discharge, and dynamic charging-discharging experiments. For the charging-redistribution experiment, the VLR model, two branch model, and three branch model results match the measurement well. The EIE model results show noticeable deviations. For the self-discharge experiment, the VLR model and EIE model are the most accurate. The three branch model results slightly mismatch the measurement during certain period of time. The two branch model is the least accurate for this experiment. For the dynamic charging-discharging experiment, the VLR model is much more accurate than the EIE model.

Chapter III analyzes supercapacitor voltage change and energy loss during charge redistribution. The VLR model for supercapacitors is used to perform this analysis. The effects of time and supercapacitor initial state on supercapacitor voltage change during charge redistribution are examined for the case where $V_1 > V_2$. For a fixed supercapacitor initial state, supercapacitor terminal voltage decreases with time because part of the charge stored in C_1 is transferred to C_2. For a particular time, supercapacitor terminal voltage drop is greater if the initial voltage difference $\Delta V = V_1 - V_2$ is larger. The impacts of time and supercapacitor initial state on supercapacitor energy loss are also studied for the case where $V_1 > V_2$. For a fixed supercapacitor initial state, the dissipated powers P_2 and P_3 due to resistors R_2 and R_3 decrease with time. The respective contributions of P_2 and P_3 to the total energy loss vary with time. The energy loss is mainly contributed by R_2 during the initial phase of charge redistribution and by R_3 for a relatively long term. The relationship between supercapacitor energy loss and supercapacitor initial state is also studied. The supercapacitor initial state is characterized in two aspects: absolute initial voltages (V_1 and V_2) and relative initial voltage difference ($\Delta V = V_1 - V_2$). During the initial phase of charge redistribution, the total dissipated power $P_2 + P_3$ and total energy loss $E_2 + E_3$ increase with relative initial voltage difference. For a relatively long term, the total dissipated power $P_2 + P_3$ and the total energy loss $E_2 + E_3$ decrease with relative initial voltage difference. The total dissipated power $P_2 + P_3$ and total energy loss $E_2 + E_3$ decrease if the absolute initial voltages decrease.

Chapter IV qualitatively studies the task scheduling problem in supercapacitor based environmentally powered wireless sensor nodes. The impacts of supercapacitor state and energy harvesting on task scheduling are investigated. A set of six simulation cases is designed to cover the various scenarios of supercapacitor state and energy harvesting to evaluate the greedy and lazy task scheduling policies in terms of two metrics: task execution and energy loss. Based on the simulation results, task scheduling rules are developed. To maximize the number of successfully executed tasks, the rules are: use the greedy schedule when supercapacitor state is $V_1 > V_2$ and energy harvesting is not available; otherwise, use the lazy policy.

Chapter V and Chapter VI quantitatively study the task scheduling problem in supercapacitor based environmentally powered wireless sensor nodes. Chapter V considers independent tasks, which are nonpreemptable tasks without precedence constraints. Chapter VI considers nonpreemptable tasks with precedence constraints.

Chapter V presents the MEDF algorithm for independent tasks. The MEDF algorithm has three steps. First, generate an initial schedule using the EDF algorithm. This step takes care of the timing constraints of tasks. Second, calculate task ready time adjustment margin based on the initial schedule. This margin determines how much delay is allowed if task ready time of the initial schedule is adjusted. Third, task ready time offset is determined based on supercapacitor state and energy harvesting. Task start time is task ready time plus the ready time offset. The MEDF algorithm takes into account the energy constraints of tasks in addition to timing constraints. The MEDF algorithm is implemented and evaluated in terms of two metrics: deadline miss rate and energy violation rate. Simulation results show that the MEDF algorithm improves the energy performance of the EDF algorithm while maintaining the same timing performance.

Chapter VI presents the MFIFO algorithm for nonpreemptable tasks with precedence constraints. The task precedence constraints are transformed into timing constraints by defining the effective release time of a task. The effective release time of a task with predecessor is equal to the maximum value between its release time and the release time of its predecessor plus the execution time of its predecessor. The MFIFO algorithm has three steps. First, create an initial schedule using the FIFO algorithm. This step takes care of the timing constraints and precedence constraints of tasks. Second, calculate the ready time adjustment margin based on the initial schedule. Third, the ready time offset is determined based on supercapacitor state and energy harvesting. The start time of a task is the ready time plus the ready time offset. The MFIFO algorithm takes into account the energy constraints of tasks in addition to the timing constraints. The MFIFO algorithm is implemented and evaluated. Simulation results show that the MFIFO algorithm improves the energy performance of the FIFO algorithm and maintains the timing performance at the same time.

The contributions have been reported in the following publications.

1. Yang, H. and Zhang, Y., “A task scheduling algorithm for supercapacitor based environmentally powered wireless sensor nodes,” *IEEE Transactions on Power Electronics*, submitted.

2. Yang, H. and Zhang, Y., “Analysis of supercapacitor energy loss for power management in environmentally powered wireless sensor nodes,” *IEEE Transactions on Power Electronics*, accepted.

3. Yang, H. and Zhang, Y., “Self-discharge analysis and characterization of supercapacitors for environmentally powered wireless sensor network applications,” *Journal of Power Sources*, vol. 196, no. 20, pp. 8866–8873, 2011.

4. Zhang, Y. and Yang, H., “Modeling and characterization of supercapacitors for wireless sensor network applications,” *Journal of Power Sources*, vol. 196, no. 8, pp. 4128–4135, 2011.

5. Yang, H. and Zhang, Y., “Modeling and analysis of a solar powered wireless sensor node,” in *Proceedings of the 2012 International Conference on Computing, Networking and Communications (ICNC 2012)*, pp. 970–974, 2012.

6. Yang, H. and Zhang, Y., “Evaluation of supercapacitor models for wireless sensor network applications,” in *Proceedings of the 5th International Conference on Signal Processing and Communication Systems (ICSPCS 2011)*, pp. 1–6, 2011.

7. Yang, H. and Zhang, Y., “An equivalent circuit model of supercapacitors for applications in wireless sensor networks,” *Proceedings of SPIE*, vol. 7981, pp. 79810D:1–79810D:9, 2011.

8. Yang, H. and Zhang, Y., “Modeling and analysis of hybrid energy storage systems for wireless sensor networks,” *Proceedings of SPIE*, vol. 7647, pp. 76472U: 1–76472U:10, 2010.

7.2 Future Work

In this dissertation, the task scheduling problem is considered for nonpreemptable tasks in a single wireless sensor node that uses supercapacitor based energy storage systems. The proposed task scheduling algorithms may be extended in three aspects.

First, the task scheduling algorithms for tasks with more characteristics can be developed. The task model used in this dissertation assumes that the tasks are nonpreemptable. For example, the MEDF and MFIFO algorithms need to be modified if task preemptivity is considered.

Second, the task scheduling problem for wireless sensor nodes using other energy storage systems can be considered. For example, hybrid energy storage systems that combine supercapacitors and rechargeable batteries have proven to be a viable configuration. In addition to the supercapacitor characteristics, the characteristics of rechargeable batteries also need to be taken into account to develop task scheduling algorithms suited for hybrid energy storage systems.

Third, the task scheduling problem for wireless sensor networks is another possible extension. The network wide task scheduling problem deserves extra efforts because different sensor nodes in a distributed network usually have different energy harvesting profiles. Meanwhile, the available energy for a single sensor node typically varies with time in a nondeterministic manner. This coupled spatial temporal uncertainty adds complexity to global power management.

REFERENCES

[1] Akyildiz, I. F., Melodia, T., and Chowdhury, K. R., "A survey on wireless multimedia sensor networks," *Computer Networks*, vol. 51, no. 4, pp. 921–960, 2007.

[2] Akyildiz, I. F., Pompili, D., and Melodia, T., "Challenges for efficient communication in underwater acoustic sensor networks," *ACM SIGBED Review*, vol. 1, no. 2, pp. 3–8, 2004.

[3] Akyildiz, I. F. and Stuntebeck, E. P., "Wireless underground sensor networks: Research challenges," *Ad Hoc Networks*, vol. 4, no. 6, pp. 669–686, 2006.

[4] Akyildiz, I., Su, W., Sankarasubramaniam, Y., and Cayirci, E., "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.

[5] Alippi, C. and Galperti, C., "An adaptive system for optimal solar energy harvesting in wireless sensor network nodes," *IEEE Transactions on Circuits and Systems I, Regular Papers*, vol. 55, no. 6, pp. 1742–1750, 2008.

[6] Belhachemi, F., Rael, S., and Davat, B., "A physical based model of power electric double-layer supercapacitors," in *Record of the 2000 IEEE Industry Applications Conference*, pp. 3069–3076, 2000.

[7] Bertrand, N., Sabatier, J., Briat, O., and Vinassa, J.-M., "Embedded fractional nonlinear supercapacitor model and its parametric estimation method," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 12, pp. 3991–4000, 2010.

[8] Brunelli, D., Moser, C., Thiele, L., and Benini, L., "Design of a solar-harvesting circuit for batteryless embedded systems," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 56, no. 11, pp. 2519–2528, 2009.

[9] Buller, S., Karden, E., Kok, D., and De Doncker, R. W., "Modeling the dynamic behavior of supercapacitors using impedance spectroscopy," *IEEE Transactions on Industry Applications*, vol. 38, no. 6, pp. 1622–1626, 2002.

[10] Burke, A., "Ultracapacitors: why, how, and where is the technology," *Journal of Power Sources*, vol. 91, no. 1, pp. 37–50, 2000.

[11] Celzard, A., Collas, F., Mareche, J., Furdin, G., and Rey, I., "Porous electrodes-based double-layer supercapacitors: pore structure versus series resistance," *Journal of Power Sources*, vol. 108, no. 1-2, pp. 153–162, 2002.

[12] Cheng, Y., "Assessments of energy capacity and energy losses of supercapacitors in fast charging-discharging cycles," *IEEE Transactions on Energy Conversion*, vol. 25, no. 1, pp. 253–261, 2010.

[13] Chong, C.-Y. and Kumar, S., "Sensor networks: evolution, opportunities, and challenges," *Proceedings of the IEEE*, vol. 91, no. 8, pp. 1247–1256, 2003.

[14] Chun, B. N., Buonadonna, P., AuYoung, A., Ng, C., Parkes, D. C., Shneidman, J., Snoeren, A. C., and Vahdat, A., "Mirage: a microeconomic resource allocation system for sensornet testbeds," in *Proceedings of the 2nd IEEE Workshop on Embedded Networked Sensors (EmNets '05)*, pp. 19–28, 2005.

[15] Conway, B. E., Pell, W., and Liu, T.-C., "Diagnostic analyses for mechanisms of self-discharge of electrochemical capacitors and batteries," *Journal of Power Sources*, vol. 65, no. 1-2, pp. 53–59, 1997.

[16] Corke, P., Valencia, P., Sikka, P., Wark, T., and Overs, L., "Long-duration solar-powered wireless sensor networks," in *Proceedings of the 4th Workshop on Embedded Networked Sensors (EmNets '07)*, pp. 33–37, 2007.

[17] Diab, Y., Venet, P., Gualous, H., and Rojat, G., "Self-discharge characterization and modeling of electrochemical capacitor used for power electronics applications," *IEEE Transactions on Power Electronics*, vol. 24, no. 2, pp. 510–517, 2009.

[18] Dutta, P., Hui, J., Jeong, J., Kim, S., Sharp, C., Taneja, J., Tolle, G., Whitehouse, K., and Culler, D., "Trio: enabling sustainable and scalable outdoor wireless sensor network deployments," in *Proceedings of the 5th International Conference on Information Processing in Sensor Networks (IPSN '06)*, pp. 407–415, 2006.

[19] El Brouji, E.-H., Briat, O., Vinassa, J.-M., Bertrand, N., and Woirgard, E., "Impact of calendar life and cycling ageing on supercapacitor performance," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 8, pp. 3917–3929, 2009.

[20] Faranda, R., "A new parameters identification procedure for simplified double layer capacitor two-branch model," *Electric Power Systems Research*, vol. 80, no. 4, pp. 363–371, 2010.

[21] Friedman, D., Heinrich, H., and Duan, D.-W., "A low-power CMOS integrated circuit for field-powered radio frequency identification tags," in *Proceedings of the 1997 IEEE International Solid-State Circuits Conference (ISSCC '97)*, pp. 294–295, 474, 1997.

[22] Handziski, V., Kopke, A., Willig, A., and Wolisz, A., "TWIST: a scalable and reconfigurable testbed for wireless indoor experiments with sensor networks," in *Proceedings of the 2nd International Workshop on Multi-hop Ad Hoc Networks: From Theory to Reality*, pp. 63–70, 2006.

[23] He, T., Krishnamurthy, S., Stankovic, J. A., Abdelzaher, T., Luo, L., Stoleru, R., Yan, T., Gu, L., Hui, J., and Krogh, B., "Energy-efficient surveillance system using wireless sensor networks," in *Proceedings of the 2nd International Conference on Mobile Systems, Applications, and Services (MobiSys '04)*, pp. 270–283, 2004.

[24] Hill, J. and Culler, D., "Mica: a wireless platform for deeply embedded networks," *IEEE Micro*, vol. 22, no. 6, pp. 12–24, 2002.

[25] Jeong, J., *A Practical Theory of Micro-Solar Power Sensor Networks.* PhD thesis, University of California at Berkeley, 2009.

[26] Jiang, X., Polastre, J., and Culler, D., "Perpetual environmentally powered sensor networks," in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN '05)*, pp. 463–468, 2005.

[27] Kansal, A., Hsu, J., Zahedi, S., and Srivastava, M. B., "Power management in energy harvesting sensor networks," *ACM Transactions on Embedded Computing Systems*, vol. 6, no. 4, pp. 32:1–32:38, 2007.

[28] Kansal, A. and Srivastava, M. B., "An environmental energy harvesting framework for sensor networks," in *Proceedings of the 2003 International Symposium on Low Power Electronics and Design (ISLPED '03)*, pp. 481–486, 2003.

[29] Kim, S. and Chou, P. H., "Size and topology optimization for supercapacitor-based sub-watt energy harvesters," *IEEE Transactions on Power Electronics*, vol. 28, no. 4, pp. 2068–2080, 2013.

[30] Kimball, J. W., Kuhn, B. T., and Balog, R. S., "A system design approach for unattended solar energy harvesting supply," *IEEE Transactions on Power Electronics*, vol. 24, no. 4, pp. 952–962, 2009.

[31] Kularatna, N., "Modern batteries and their management: Part 1," in *Proceedings of the 36th Annual Conference on IEEE Industrial Electronics Society (IECON '10)*, pp. 1–103, 2010.

[32] Lai, J.-S., Levy, S., and Rose, M., "High energy density double-layer capacitors for energy storage applications," *IEEE Aerospace and Electronic Systems Magazine*, vol. 7, no. 4, pp. 14–19, 1992.

[33] Lajnef, W., Vinassa, J. M., Briat, O., Azzopardi, S., and Woirgard, E., "Characterization methods and modelling of ultracapacitors for use as peak power sources," *Journal of Power Sources*, vol. 168, no. 2, pp. 553–560, 2007.

[34] Le, T., Mayaram, K., and Fiez, T., "Efficient far-field radio frequency energy harvesting for passively powered sensor networks," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 5, pp. 1287–1302, 2008.

[35] Li, Q., Aslam, J., and Rus, D., "Online power-aware routing in wireless adhoc networks," in *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking (MobiCom '01)*, pp. 97–107, 2001.

[36] Linear, "LTC3401 Datasheet." http://cds.linear.com/docs/Datasheet/3401fb.pdf.

[37] Liu, J. W. S., *Real-Time Systems*. Prentice Hall, 2000.

[38] Liu, S., Lu, J., Wu, Q., and Qiu, Q., "Harvesting-aware power management for real-time systems with renewable energy," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 20, no. 8, pp. 1473–1486, 2012.

[39] Liu, T., Pell, W., and Conway, B., "Self-discharge and potential recovery phenomena at thermally and electrochemically prepared RuO_2 supercapacitor electrodes," *Electrochimica Acta*, vol. 42, no. 23-24, pp. 3541–3552, 1997.

[40] Lorincz, K., Malan, D., Fulford-Jones, T., Nawoj, A., Clavel, A., Shnayder, V., Mainland, G., Welsh, M., and Moulton, S., "Sensor networks for emergency response: challenges and opportunities," *IEEE Pervasive Computing*, vol. 3, no. 4, pp. 16–23, 2004.

[41] Maxwell, "BCAP0010 Datasheet." http://www.maxwell.com/products/ultra capacitors/docs/03152013dshcseries.pdf.

[42] Meninger, S., Mur-Miranda, J. O., Amirtharajah, R., Chandrakasan, A. P., and Lang, J. H., "Vibration-to-electric energy conversion," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 9, no. 1, pp. 64–76, 2001.

[43] Min, R., Bhardwaj, M., Cho, S.-H., Sinha, A., Shih, E., Wang, A., and Chandrakasan, A., "An architecture for a power-aware distributed microsensor node," in *Proceedings of the 2000 IEEE Workshop on Signal Processing Systems (SiPS '00)*, pp. 581–590, 2000.

[44] Moser, C., *Power management in energy harvesting embedded systems*. PhD thesis, Swiss Federal Institute of Technology, 2009.

[45] Moser, C., Chen, J.-J., and Thiele, L., "An energy management framework for energy harvesting embedded systems," *ACM Journal on Emerging Technologies in Computing Systems*, vol. 6, no. 2, pp. 7:1–7:21, 2010.

[46] Moteiv, "TmoteSky Datasheet." http://www.eecs.harvard.edu/konrad/projects /shimmer/references/tmote-sky-datasheet.pdf.

[47] Niu, J., Conway, B. E., and Pell, W. G., "Comparative studies of self-discharge by potential decay and float-current measurements at C double-layer capacitor and battery electrodes," *Journal of Power Sources*, vol. 135, no. 1-2, pp. 332–343, 2004.

[48] Ongaro, F., Saggini, S., and Mattavelli, P., "Li-ion battery-supercapacitor hybrid storage system for a long lifetime, photovoltaic-based wireless sensor network," *IEEE Transactions on Power Electronics*, vol. 27, no. 9, pp. 3944–3952, 2012.

[49] Pandolfo, A. and Hollenkamp, A., "Carbon properties and their role in supercapacitors," *Journal of Power Sources*, vol. 157, no. 1, pp. 11–27, 2006.

[50] Polastre, J., Szewczyk, R., and Culler, D., "Telos: Enabling ultra-low power wireless research," in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN '05)*, pp. 364–369, 2005.

[51] Rabaey, J., Ammer, J., Karalar, T., Li, S., Otis, B., Sheets, M., and Tuan, T., "PicoRadios for wireless sensor networks: the next challenge in ultra-low power design," in *Proceedings of the 2002 IEEE International Solid-State Circuits Conference (ISSCC '02)*, pp. 200–201, 2002.

[52] Rafik, F., Gualous, H., Gallay, R., Crausaz, A., and Berthon, A., "Frequency, thermal and voltage supercapacitor characterization and modeling," *Journal of Power Sources*, vol. 165, no. 2, pp. 928–934, 2007.

[53] Raghunathan, V., Schurgers, C., Park, S., and Srivastava, M., "Energy-aware wireless microsensor networks," *IEEE Signal Processing Magazine*, vol. 19, no. 2, pp. 40–50, 2002.

[54] Raghunathan, V., Kansal, A., Hsu, J., Friedman, J., and Srivastava, M., "Design considerations for solar energy harvesting wireless embedded systems," in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN '05)*, pp. 457–462, 2005.

[55] Ricketts, B. and Ton-That, C., "Self-discharge of carbon-based supercapacitors with organic electrolytes," *Journal of Power Sources*, vol. 89, no. 1, pp. 64–69, 2000.

[56] Riu, D., Retiere, N., and Linzen, D., "Half-order modelling of supercapacitors," in *Record of the 2004 IEEE Industry Applications Conference*, pp. 2550–2554, 2004.

[57] Rong, P. and Pedram, M., "Extending the lifetime of a network of battery-powered mobile devices by remote processing: a markovian decision-based approach," in *Proceedings of the 40th Annual Design Automation Conference (DAC '03)*, pp. 906–911, 2003.

[58] Roundy, S., Steingart, D., Frechette, L., Wright, P., and Rabaey, J., "Power sources for wireless sensor networks," in *Wireless Sensor Networks* (Karl, H., Wolisz, A., and Willig, A., eds.), vol. 2920 of *Lecture Notes in Computer Science*, pp. 1–17, 2004.

[59] Shah, R. and Rabaey, J., "Energy aware routing for low energy ad hoc sensor networks," in *Proceedings of the 2002 IEEE Wireless Communications and Networking Conference (WCNC '02)*, pp. 350–355, 2002.

[60] Shenck, N. and Paradiso, J., "Energy scavenging with shoe-mounted piezoelectrics," *IEEE Micro*, vol. 21, no. 3, pp. 30–42, 2001.

[61] Sikha, G., White, R. E., and Popov, B. N., "A mathematical model for a lithium-ion battery/electrochemical capacitor hybrid system," *Journal of The Electrochemical Society*, vol. 152, no. 8, pp. A1682–A1693, 2005.

[62] Simjee, F. and Chou, P. H., "Everlast: Long-life, supercapacitor-operated wireless sensor node," in *Proceedings of the 2006 International Symposium on Low Power Electronics and Design (ISLPED '06)*, pp. 197–202, 2006.

[63] Singh, S., Woo, M., and Raghavendra, C. S., "Power-aware routing in mobile ad hoc networks," in *Proceedings of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom '98)*, pp. 181–190, 1998.

[64] Sinha, A. and Chandrakasan, A., "Dynamic power management in wireless sensor networks," *IEEE Design & Test of Computers*, vol. 18, no. 2, pp. 62–74, 2001.

[65] Song, H.-K., Hwang, H.-Y., Lee, K.-H., and Dao, L. H., "The effect of pore size distribution on the frequency dispersion of porous electrodes," *Electrochimica Acta*, vol. 45, no. 14, pp. 2241–2257, 2000.

[66] Stordeur, M. and Stark, I., "Low power thermoelectric generator-self-sufficient energy supply for micro systems," in *Proceedings of the XVI International Conference on Thermoelectrics (ICT '97)*, pp. 575–577, 1997.

[67] Sudevalayam, S. and Kulkarni, P., "Energy harvesting sensor nodes: Survey and implications," *IEEE Communications Surveys & Tutorials*, vol. 13, no. 3, pp. 443–461, 2011.

[68] Szewczyk, R., Mainwaring, A., Polastre, J., Anderson, J., and Culler, D., "An analysis of a large scale habitat monitoring application," in *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems (SenSys '04)*, pp. 214–226, 2004.

[69] Tan, Y. K. and Panda, S. K., "A novel piezoelectric based wind energy harvester for low-power autonomous wind speed sensor," in *Proceedings of the 33rd Annual Conference of the IEEE Industrial Electronics Society (IECON '07)*, pp. 2175–2180, 2007.

[70] Taneja, J., Jeong, J., and Culler, D., "Design, modeling, and capacity planning for micro-solar power sensor networks," in *Proceedings of the 7th International Conference on Information Processing in Sensor Networks (IPSN '08)*, pp. 407–418, 2008.

[71] Vigorito, C., Ganesan, D., and Barto, A., "Adaptive control of duty cycling in energy-harvesting wireless sensor networks," in *Proceedings of the 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON '07)*, pp. 21–30, 2007.

[72] Voigt, T., Ritter, H., and Schiller, J., "Utilizing solar power in wireless sensor networks," in *Proceedings of the 28th Annual IEEE International Conference on Local Computer Networks (LCN '03)*, pp. 416–422, 2003.

[73] Warneke, B., Atwood, B., and Pister, K., "Smart Dust mote forerunners," in *Proceedings of the 14th IEEE International Conference on Micro Electro Mechanical Systems (MEMS '01)*, pp. 357–360, 2001.

[74] Werner-Allen, G., Swieskowski, P., and Welsh, M., "MoteLab: a wireless sensor network testbed," in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN '05)*, pp. 483–488, 2005.

[75] Xu, Y., Heidemann, J., and Estrin, D., "Geography-informed energy conservation for ad hoc routing," in *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking (MobiCom '01)*, pp. 70–84, 2001.

[76] Younis, M., Youssef, M., and Arisha, K., "Energy-aware routing in cluster-based sensor networks," in *Proceedings of the 10th IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunications Systems (MASCOTS '02)*, pp. 129–136, 2002.

[77] Zhang, P., Sadler, C. M., Lyon, S. A., and Martonosi, M., "Hardware design experiences in ZebraNet," in *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems (SenSys '04)*, pp. 227–238, 2004.

[78] Zhao, Y., Govindan, R., and Estrin, D., "Residual energy scan for monitoring sensor networks," in *Proceedings of the 2002 IEEE Wireless Communications and Networking Conference (WCNC '02)*, pp. 356–362, 2002.

[79] Zhu, T., Zhong, Z., Gu, Y., He, T., and Zhang, Z.-L., "Leakage-aware energy synchronization for wireless sensor networks," in *Proceedings of the 7th International Conference on Mobile Systems, Applications, and Services (MobiSys '09)*, pp. 319–332, 2009.

[80] Zubieta, L. and Bonert, R., "Characterization of double-layer capacitors for power electronics applications," *IEEE Transactions on Industry Applications*, vol. 36, no. 1, pp. 199–205, 2000.