Adaptive Battery Charge Scheduling with Bursty Workloads

Dylan Lexie , Shan Lin, and Jie Wu
Department of Computer and Information Sciences
Temple University
dylan.lexie, shan.lin, jie.wu@temple.edu

***Abstract*—Battery-powered wireless sensor devices need to be charged to provide the desired functionality after deployment. Task or even device failures can occur if the voltage of the battery is low. It is very important to schedule the recharge of batteries in time. Existing battery scheduling algorithms usually charge a battery when its voltage drops below a fixed level. Such algorithms work well when the workloads are predictable. However, workloads of wireless sensors can be highly bursty, i.e., extensive sensing and communication tasks usually occur in a very short time period. If such a bursty workload occurs when the battery voltage is low, the battery energy can be depleted very quickly, resulting in system task failures before the device can be recharged. To deal with unpredictable bursty workloads, we investigate battery characteristics with different workloads via experiments. Based on the empirical results, we build an adaptive linear model and propose a feedback control based battery charge scheduling algorithm. This algorithm dynamically adjusts the battery charge threshold for recharge scheduling, adapting to bursty workloads. We have tested our algorithms in extensive simulations with traces obtained from real experiments. Evaluation results show that our algorithms can adapt to bursty workloads. Compared to existing algorithms, our algorithm achieves a 68.26% lower task failure ratio with a 3.45% sacrifice on system lifetime under bursty workloads.**

***Index Terms*—battery, burstiness, scheduling, control, energy efficiency**

I. Introduction

With the advance of wireless and sensing technologies, wireless sensor networks are deployed for various applications, such as military surveillance, scientific exploration, and environmental monitoring. These wireless sensor devices are usually powered by rechargeable batteries. To maintain desired functionality of such devices, it is very important to charge their batteries in time after deployment. However, in real systems on-demand battery charging scheme is widely used, since system users may not be aware of the low power battery status until nodes start to fail. Moreover, it requires extra time and effort to recharge the battery after deployment.

Battery characteristics and scheduling have been studied extensively [1–6]. These studies have provided valuable research results. A large part of these works assume stable or predictable workloads. However, the workloads for wireless sensor networks are usually dynamic due to the nature of physical phenomena they monitor. Therefore, wireless sensor nodes may consume energy at a highly dynamic rate. Recent studies [7–9] suggest that the energy usage of these devices can be bursty i.e., a device consumes a large amount of energy within a short period of time due to an extensive workload. Such bursty energy usage can deplete a battery very quickly when its voltage is already low. As shown in Figure 1, if a bursty workload occurs when the battery is close to depletion, it causes a dramatic discharge in a short time period. When the battery's voltage drops out of normal operational voltage levels, tasks and even the system fail. We call this problem *dramatic discharge with bursty workloads*. It is very challenging for battery charge schedules to deal with bursty workloads.

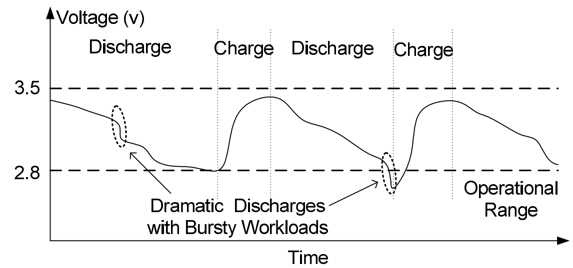
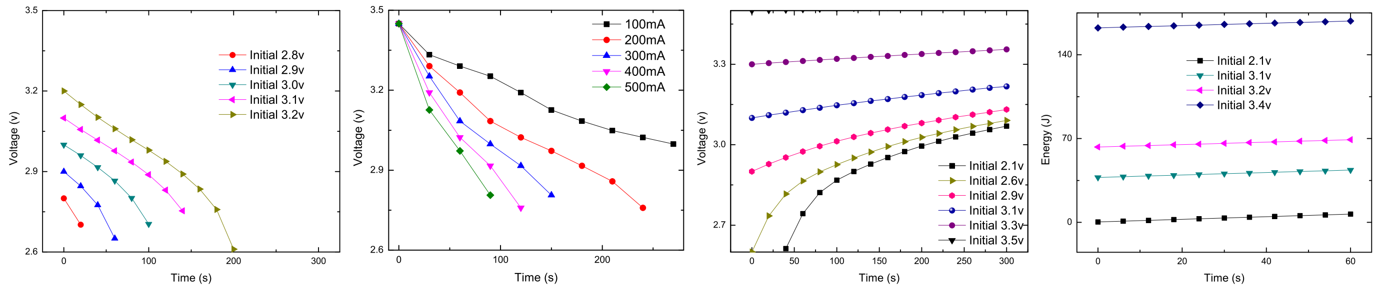


Fig. 1: Battery Charge with Bursty Load

With simplified workload assumptions, many of existing battery charge scheduling algorithms use fixed schedules to charge the battery; other on-demand algorithms require the battery to be charged when its voltage drops below a fixed level. These schemes work well when the power consumption doesn't vary dramatically. Unfortunately, they all suffer from the dramatic discharge with bursty loads. In this work, we have demonstrated that those existing recharging schemes 1) do not fully utilize energy that is stored in batteries if batteries are charged conservatively, and 2) do not sustain desired services of devices if batteries are charged aggressively.

In order to achieve efficient usage of battery energy and to sustain desired services for as long as possible, it is essential to dynamically schedule the battery charge process based on usage patterns. Recent research [10] demonstrates some potential designs for dynamic battery scheduling algorithms. In this work, we investigate the impact of bursty workloads on system lifetime via experiments. Based on our experimental results, we create an empirical model of battery lifetime under different workloads. With this empirical model, we propose feedback control based adaptive schedules for charging the battery. Our algorithm design aims to optimize the lifetime of a device and to reduce the number of task misses with a fixed number of battery charges. The rechargeable batteries have limited recharge cycles due to cyclic memory and crystalline formation [11]. Specifically, our online algorithm estimates



(a) Discharging Rates under Different Voltages (b) Discharging Rates under Different Currents (c) Charging Rates under Different Voltages (d) Total Energy Absorption under Different Voltages

Fig. 2: Battery Characteristics

the remaining lifetime of a device based on the discharging rate of its batteries and its past workload. A markov model is employed to approximate the bursty workload. Our algorithm demonstrates a good tradeoff between the system performance and the system lifetime.

The process of energy consumption and recovery is affected by many uncertainties. Intuitively, if all of the energy that is produced and consumed is known at every time point, we can find the optimal battery charge schedule. However, the workload could be unpredictable. In order to achieve a stable system performance, we also develop robust charge schedules.

Our dynamic battery charge scheduling algorithms are evaluated in extensive simulations. Simulation results show that our feedback based dynamic schedules achieve a 68.26% lower task failure ratio compared to existing schemes, with merely a 3.45% decrease in system lifetime. Also, our robust battery charge schedule can achieve even lower task failure ratios with bursty workloads.

The rest of this paper is organized as follows. Our motivation is presented in Section II. In Section III, we show the battery charge scheduling algorithm. In Section IV, our solution is evaluated in simulation. State-of-the-art is described in Section V. Conclusions of this work are stated in Section VI.

II. An Empirical Study on Battery Charateristics

In order to design better battery charging schedules for dealing with bursty workloads, we need to understand a battery's charging and discharging characteristics. In this section, we conducted several sets of experiments to study battery charateristics under different workloads. These experiments are repeated with multiple different batteries. The data collected from different batteries is similar. We used five UltraFire brand AA lithium ion batteries with a maximum voltage rating of 3.6v and a nominal charge capacity of 900mAh in these experiments.

In the first set of experiments, we discharged a battery with a constant workload. We measured and recorded its voltage changes. The data obtained from one battery is plotted in Figure 2 (a). The curves in Figure 2 (a) represent the battery discharging rates from six starting voltage levels for durations of five minutes. From Figure 2 (a), we notice that the battery voltage decreases through the entire discharge process. These curves are approximately linear. The voltage decreases relatively slowly when the voltage is high and relatively quickly when the voltage is low. This data can help us build a model of a battery's voltage over time with a fixed workload.

In the second set of experiments, we discharge a battery multiple times with different workloads. For each experiment, we use a constant workload, represented by a constant current. We measured and recorded its voltage change. The data obtained is plotted in Figure 2 (b). Each curve in this figure represents a battery's discharging rate with a specific workload. From this figure, we can see that all of these figures are approximately linear in the battery's operational range, specifically if the voltage is higher than 2.8 volts. However, the slopes of these curves are significantly different. These curves are dramatically steeper with higher workloads, which conforms to the rate capacity effect [12]. This experimental data helps us to capture the relation between a battery's discharging rate and its specific workload.

In the third set of experiments, we charged a battery from a uniform source voltage of five volts. We measured and recorded battery voltage during the charging process. The data obtained is plotted in Figure 2 (c). Each curve in Figure 2 (c) shows the charging rate of the battery from one starting voltage. We notice that the charging rates with different starting voltages are very similar. The voltage increases a little bit more quickly while it is low, and the rate becomes stable as the voltage rises. For this charging experiment, we also computed the total amount of energy that is absorbed by the battery. Figure 2 (d) shows the total absorbed energy as a function of time. Similarly, we can see the battery's energy storage increases slowly at an approximately constant rate.

III. Design of Dynamic Charge Schedule

In this section, we describe a battery model based on our extensive empirical studies. With this battery model, we design a feedback control based battery charge scheduling algorithm. Our algorithm aims to sustain a high quality system performance by adapting to bursty workloads.

A. Battery and Workload Models

From our experimental results, we create an empirical battery energy model. This model characterizes the battery discharging rate with different workloads.

The battery discharging charateristics discussed in last section suggest an approximate linear relation relation between battery discharging rate and workload:

$$r_{discharge} = a \cdot w + b \tag{1}$$

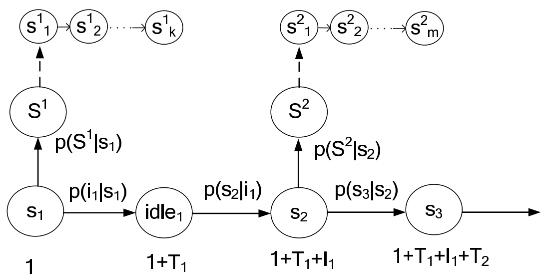


Fig. 3: HMM for Bursty Task

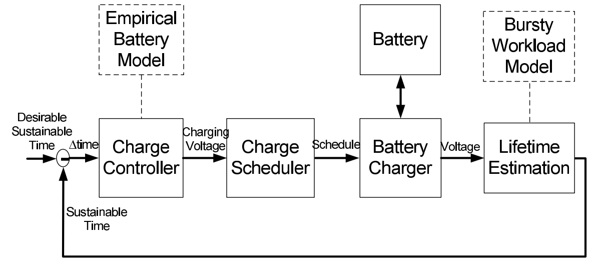


Fig. 4: Battery Charge Control Loop

where $r_{discharge}$ represents the battery discharging rate, w represents system workload, a and b are model parameters that are different with various types of batteries and can change over the lifetime of the battery. These parameters can be obtained via online system identification using the least square approximation. This battery model allows us to predict battery lifetime based on workloads and adjust the charging process accordingly.

The basic task models are described as follows. The device has a set of tasks S to execute. Each task s_i in this set S must run for a certain period of time T_i; if a task is stopped within its executing time period, it fails. When the system is not executing any task, its state is idle. The time duration for the idle state i is represented as I_i. The power consumption of a task s_i is represented as j_i.

Here we consider three types of tasks: 1) stochastic tasks, 2) bursty tasks, and 3) a hybrid of the first two types. Stochastic tasks occur randomly; their occurrences are independent of one another. Bursty tasks are highly correlated in time. Their occurrences usually happen in the same time period. We note that the goal of this task model is to describe the burstiness of energy imposed by system workloads. Application level couplings of tasks are not the focus of this model.

We model these bursty tasks with a hidden markov model (HMM). As shown in Figure 3, a group of tasks represented by a specific subset S^j of S usually run together. A task in this group is represented as ${s^j}_i$. The transition probability from a normal task s_i to a burst of tasks S^j is represented by $p(S^j|s_i)$. The transition probability from a normal task s_i to another task, say s_k, is represented by $p(s_k|s_i)$. $idle$ represents a type of low-power task.

The total energy consumption of such a burst can be represented by the following equation.

$$E(S^j) = \sum^{s_i \in S^j} j_i \tag{2}$$

The maximum amount of energy consumption of any burst is represented by E_{max}.

$$E_{max} = \max\{E(S^j)\} \tag{3}$$

B. Adaptive Charge Schedules

Battery charge schedules determine the time to charge a battery. Intuitively, a battery should be charged as late as possible for a longer system lifetime. The system lifetime can be defined as follows. $optime$ represents the operation time after the ith recharge. K is the maximum number of recharges the battery can handle.

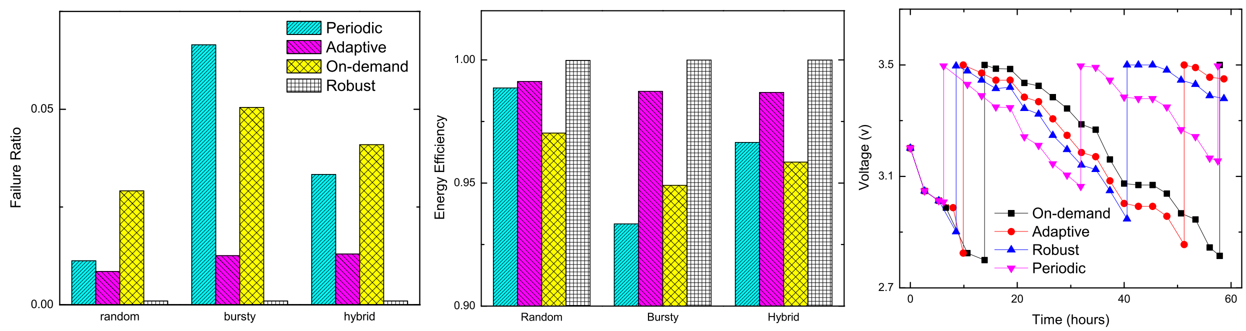
$$lifetime = \sum_{i=1}^{K} optime_i \tag{4}$$

Charging the battery as late as possible can also achieve higher energy efficiency since the battery energy is used completely after each charge. However, if the battery is deeply depleted, when a bursty workload occurs, the battery will run out of power immediately. As a result, the tasks fail and the system performance decreases. Therefore, we need to find a good tradeoff between the energy efficiency and the system performance in our charging algorithm.

As the system performance, in terms of the task failure ratio, is a central metric for system service quality, we define the system *service sustainability* as the length of time during which no tasks fail. The system sustainability can be defined as follows. $n_{failure}(i)$ is the number of task failures during the ith charging period.

$$\begin{gathered} sustainability = \sum_{i=1}^{k} optime_i \\ \text{subject to} \sum_{i=1}^{k} n_{failure}(i) = 0 \end{gathered} \tag{5}$$

Our goal is to design a battery charging algorithm that optimizes the system sustainability and achieves high charging efficiency. It is very challenging to achieve this goal, especially when bursty tasks are considered. Combined with dynamic battery characteristics, such as the rate capacity effect, it is even more difficult to predict the optimal charging time. The rate capacity effect indicates that a bursty load can significantly decrease the deliverable energy of the battery. The reason is that the battery discharging rate affects the deliverable energy of the battery significantly. If a task that requires a relatively high current comes up, the battery may not be able to supply the energy required, within the time required, to complete this task, even though it might be able to handle a task that takes longer to execute but requires the same amount of energy. Compared with previous charging algorithms, which usually use a fixed level of remaining energy, we design a feedback based algorithm that dynamically adjusts charging schedules. The feedback block diagram is shown in Figure 4. The control input is the difference between the desirable sustainable lifetime and the expected sustainable lifetime. The control output is the charging level threshold in voltage. The charge controller uses the empirical battery model to adjust the charging level. The charging level is used by the charge scheduler to select batteries to charge if their current voltages drop below this charging level. The Battery Charger charges batteries according to the specified schedules from the Charge



(a) Task Failure Ratio (b) Energy Efficiency (c) A Snapshot of Battery Cycles

Fig. 5: Charge Scheduling Algorithms with Different Workloads

Scheduler. The life estimation module estimates the expected sustainable lifetime, based on our bursty workload model.

If the current workload is s_i, and the task model indicates a high transition probability $p(S^j|s_i)$, the life estimation model will calculate the expected lifetime $T(t)_{expsus}$ based on the following equation:

$$T(t)_{expsus} = (v_c - v_o)/r_{discharge} \tag{6}$$

where v_c is the current voltage, and v_o is the lower bound of the device's operational voltage. The discharging rate $r_{discharge}$ is calcuated based on our empirical model:

$$r_{discharge} = a \cdot w + b = a \cdot (E(S^j)/T_{S^j}) + b \tag{7}$$

where $E(S^j)$ and T_{S^j} are the total energy consumption and the time duration of the bursty workload S^j. The estimated lifetime is compared against the specified sustainable lifetime T_{sus} to obtain a control error in sustainability $e(t)_{sus}$:

$$e(t)_{sus} = T_{sus} - T(t)_{expsus} \tag{8}$$

Here, T_{sus} represents the amount of time required to setup the next recharge, for example, 5 minutes for a mobile device. It can be specified by the user based on the application and system requirements. The control algorithm is triggered when the expected sustainable time is close to the specified sustainable time so that the charging schedule can be adjusted in time. When battery voltage is high ($T_{sus} << T(t)_{expsus}$), the control algorithm is not triggered.

Upon obtaining errors of sustainability, the charge controller calculates the starting level in a short adjustment period before the next battery charge. The charge controller makes adjustments on the starting level for the battery charge schedule when: 1) the expected sustainability is much larger than the desirable sustainability, and 2) the expected sustainability is dramatically smaller than the desirable sustainability. Also, based on the empirical battery models that we obtained from the experiments, we use a P control to make adjustments on starting charge levels as follows.

$$v_t(t+1) = v_t(t) + K_p \cdot r_{discharge}(t) \cdot e(t)_{sus} \tag{9}$$

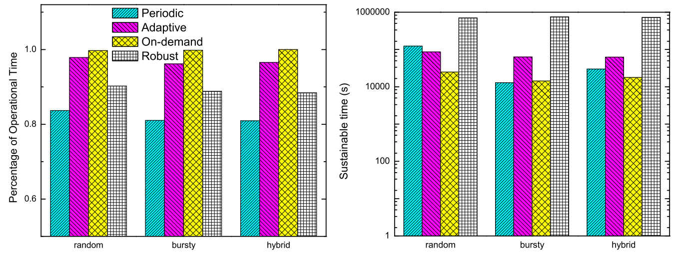
Here, $v_t(t)$ represents the starting charge level at time t, and K_p represents the proportional gain. On one hand, if the expected sustainability is lower than the desirable value, the system may suffer from task failures when a burst in the workload arrives. In this case, the starting charge level should be increased so that batteries can be charged early enough to deal with the burst. On the other hand, if the expected sustainability is higher than the desirable value, the system may have more than a sufficient amount of energy for the incoming workload burst. In this case, the starting charge level will be decreased, so that batteries can be charged later to achieve better efficiency without experiencing task failures. Since we use the linear battery models, it is necessary to use a proportional control design to gradually adjust the charging threshold to adapt. Before each battery charge, this feedback based design runs a few loops discretely until it converges.

To better deal with dynamic battery models and workloads, we have designed a robust control solution for battery charging based on worst case bursty workload estimations. The robust charging strategy suggests that the battery should be recharged as soon as the device cannot support the maximal bursty load. In other words, the remaining energy is less or equal to E_{max}. This maximal bursty load is estimated based on the workload history. As long as no bursty load with a higher energy consumption occurs in the future, this condition guarantees no task failures and achieves the longest sustainability. However, it is not efficient since in most cases the worst case bursty load may not occur at the moment that the battery is charged. As a result, this conservative strategy causes a large amount of the remaining energy E_{max} to be left unused in the battery most of the time.

IV. Evaluation

We set up a series of simulations based on collected experimental data. After recording the current and the voltage across multiple battery charging cycles from full to depleted, we build a voltage based discharge table. An abstract virtual battery containing voltage and energy information is subjected to a series of simulated tasks, represented by periods of uniform power consumption. The battery's energy is drained linearly during individual tasks, and the voltage is updated accordingly at regular intervals as well as after each task completes: after subtracting the consumed energy from the virtual batterys energy supply, we match the battery's energy value in the discharge table and linearly interpolate the corresponding voltage entries. Initial energy is determined on the basis of a specified initial voltage of 3.5v. The lower bound for operational battery voltage level is set to 2.8v.

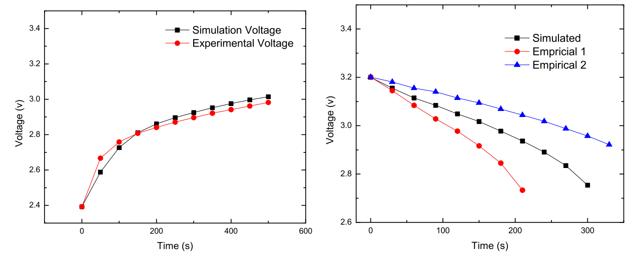
We have implemented two existing charging schemes: an on demand schedule and a periodic schedule. The on-demand



(a) System Lifetime

(b) Average Sustainability

Fig. 6: System Lifetime and Sustainability



(a) Charging Validation

(b) Discharging Validation

Fig. 7: Simulation Validation

schedule (On-demand) uses a fixed voltage level to trigger battery charges. The periodic schedule (Periodic) uses a fixed time interval to recharge the battery. Our control-based schedule (Adaptive) is also implemented. We tuned the controller and set the proportional gain as 0.2. The sampling period is the same as our simulation elapsed time. We then implemented the robust control solution (Robust). We tested these three algorithms under three types of workloads: random, bursty, and hybrid. We randomly select some random tasks to aggregate them into the bursty tasks, then we assign values for the corresponding markov models. For the hybrid workload, we set 66.67% of the tasks to be bursty. The rest of the tasks are random tasks. Our task model is trained to learn the bursty workloads in each run as our experiments progress. The number of charging cycles is set as 10,000. We ran our simulations 40 times to obtain statistical results.

The evaluation results are shown in Figure 5. We plot the task failure ratios of these four algorithms under three types of workloads in Figure 5 (a). From this figure, we have three observations. First, the task failure ratios of four algorithms are all below 10% with random workloads. This is because the energy consumption of random workloads is evenly distributed over time. The adjustments made by Adaptive and Robust are helpful but not significant. Second, the task failure ratios of Adaptive and Robust are significantly lower than On-demand and Periodic under bursty and hybrid workloads. The task failure ratio of Robust is close to zero. Finally, we can see that the task failure ratios of Adaptive and Robust under hybrid workloads are similarly good with those under random and bursty workloads. As the hybrid workloads are the most realistic, it demonstrates that Adaptive and Robust can work well in reality.

To better demonstrate the impact of bursty tasks on our algorithms, we take a closer look at these four algorithms' performances in Figure 5 (c). The curves in this figure demonstrate the voltage changes of these algorithms during two charging cycles. We can see that bursty tasks can cause dramatic energy consumption at about the 50 hour mark, which lead to the task failures of On-demand, as the battery voltage is close to the operational bound while this burst occurs. Whereas for Adaptive, the begining of bursty tasks can be detected, and a battery charge can be triggered early enough to avoid failures. Some failures still occur (1.30%) for Adaptive due to inaccurate estimations. For Robust, the failure ratio is very low (0.01%) since it uses the worst case estimation to select a very conservative charging level. However, the battery cycle of Robust is much shorter than that of Adaptive.

Figure 5 (b) shows the comparison of energy efficiencies of these four algorithms. The energy efficiency is calculated based on Equation 10.

$$efficiency = E_{completed\_tasks}/E_{total} \tag{10}$$

We use $E_{completed\_tasks}$ to represent the amount of energy consumed on tasks that are completed; E_{total} represents the total amount of energy consumed in the battery's lifetime. From this figure, we can see that Robust achieves the highest energy efficiency in all cases. This is because Robust has very few task failures: 99.99% of the energy consumed is used to execute tasks successfully. Adaptive achieves a stable energy efficiency of about 98.68% with different workloads, which is short of Robust but much better than Periodic and On-demand. The reason is that Adaptive has very low task failure rates, and the energy wasted on uncompleted tasks is also very low. Whereas the energy efficiencies for both Periodic and On-demand are much lower, especially with bursty workloads. As they do not adapt to the dramatic workload variation, both Periodic and On-demand waste a small amount of energy on tasks that they could not complete.

Based on the experimental data, we further calculated the system lifetime and system sustainability. Figure 6 (a) shows the system lifetime of the four algorithms with a fixed number of recharges (10,000). We use the on-demand algorithm as a baseline, and we calculate the ratio of lifetime of the other three algorithms and plot them on this figure. From this figure, we can see that On-demand achieves the longest lifetime with all workloads, as it does not recharge until tasks start to fail or the battery is deeply depleted. Both Adaptive and On-demand achieve a very long system lifetime, 96.55-97.86% and 99.72-100.00%, respectively. Adaptive's lifetime is 9.17% higher than Robust, as Adaptive does not significantly change the charging schedule; it only adaptively brings forward the recharge when a burst is detected and postpones the recharge when no bursts are coming. Robust uses a conservative recharging voltage level, which is why its total amount of operational time is 8.41% below Adaptive.

Based on the experimental data, we further calculated the average sustainability of four algorithms, according to Equation 5, and plot them in Figure 6 (b). From this figure, we can see that Robust achieves the longest sustainability with all three workloads. The reason is that Robust keeps track of the largest amount of energy consumption of a burst. The system can continue running without a failure for a very long time until a larger burst can occur. In regular workloads, worst case bursts do not occur often. Although lower than Robust, Adaptive achieves much higher sustainability than On-demand and Periodic in our log scale figure. Adaptive still has task failures since the probability-based bursty task estimation is not perfect. Both On-demand and Robust have low sustainability due to frequent task failures and their passive charging schedules.

To validate our simulation results, we have simulated a set of experiments with data traces from real experiments. Figure 7 (a) compares the charging results of the simulation and the real experiment. We can see an appropriate match between simulated and real battery voltage over time. Figure 7 (b) compares the discharging results in simulation and in two real experiments. A resistor circuit (equivalent to a resistance of 22.4 ohms) is used. The experimental batteries are different but have the same manufacturers specifications: 3.6v AA Li-ion batteries rated at 900mAh. The virtual battery's curve is like-shaped and has values only inside the variations of these real batteries. These two figures demonstrate that our simulations are consistent with real experiments.

Overall, we can see that 1) existing static battery charge algorithms with fixed recharging conditions, represented by On-demand and Periodic, do not deal with bursty tasks very well. They suffer from task failures and do not provide stable service quality over time; whereas the feedback based dynamic schedules achieve a 68.26% lower task failure ratio, with a slight sacrifice on system lifetime (3.45%) and sustainable time than existing algorithms. 2) Robust produces conservative schedules, which greatly reduce the task failure ratios based on the past worst case. Adaptive dynamically changes charge schedules based on recent bursty workload estimations. 3) Robust is suitable for applications that do not tolerate task failures, while energy and lifetime are not major concerns. Adaptive is better for systems that need a high task success ratio and also a long system lifetime with a bursty workload.

V. State of the Art

Related research areas include battery system designs and battery scheduling algorithms. Battery system designs have been studied extensively in recent decades. Research results in this field have boosted energy efficiency significantly. Authors of [13] studied the impact of battery life and capacity on low power system designs. [14] presents a special battery system design for hybrid electric vehicles. In [4], authors design mathematical models to capture battery discharge characteristics. Our focus is on the battery charge scheduling algorithms, which can be integrated into these system platforms.

Research works on battery modeling and scheduling algorithms provided a solid thereoretical foundation for battery-based systems. Authors of [5] describe the subject of battery management systems and provide detailed models. In [6], authors present an electrical circuit level battery model for performance prediction. In [1], authors describe a scheduling algorithm for battery discharging that uses the charge recovery mechanism without any additional delay in supplying the required power. A recent study [15] investigates battery dynamics, especially with varying discharge currents. Our work present a new battery model for lifetime prediction as well as a feedback control based scheduling algorithm for battery discharging. In [10], authors design a battery-based energy scavenging system for low-power mobile sensor platforms. These works usually consider a regular random workload. In [16], authors propose an approach to dynamically balance battery conservation and application quality by monitoring the energy supply and demand and by maintaining a history of application energy use. Authors of [2, 3] have designed a battery system for vehicles and proposed dynamic configuration, charging, and discharging algorithms. These works consider a regular workload. Our work investigates the impact of bursty discharging processes on battery performance and proposes a control based scheduling algorithm for recharging the battery.

VI. Conclusion

Existing battery scheduling algorithms usually charge a battery when its voltage level drops below a fixed level. Those algorithms work well with stochastic workloads. However, on battery-powered wireless sensor nodes, extensive workloads may occur in a very short time period. When such highly bursty workloads are considered, the existing algorithms are not effective or efficient. To maintain a high system performance and to increase energy efficiency, we designed a feedback control based scheduling algorithm. Our algorithm is based on an empirical battery model obtained from experiments. Adapting to workload and battery characterisitics, this algorithm achieves a 68.26% lower task failure ratio, with a decrease of 3.45% in the system lifetime under bursty workloads, compared to existing algorithms.

References

[1] C. Chiasserini, R. R. Rao, and S. Member, "Energy efficient battery management," in *Proc. of Infocom*, 2001.

[2] H. Kim and K. G. Shin, "Scheduling of battery charge, discharge, and rest," in *Proc. of the RTSS*, 2009.

[3] ——, "On dynamic reconfiguration of a large-scale battery system," in *Proc. of the RTAS*, 2009.

[4] R. Rao, S. Vrudhula, and D. N. Rakhmatov, "Battery modeling for energy-aware system design," *Computer*, 2003.

[5] H. J. Bergveld, "Battery management systems : design by modelling," Ph.D. dissertation, Enschede, June 2001.

[6] M. Chen and G. A. Rincon-Mora, "Accurate electrical battery model capable of predicting runtime and iv performance," *IEEE TEC*, 2006.

[7] M. H. Alizai, O. Landsiedel, J. A. B. Link, S. Götz, and K. Wehrle, "Bursty traffic over bursty links," in *Proceedings of the SenSys '09*.

[8] T. He, S. Krishnamurthy, L. Luo, T. Yan, L. Gu, R. Stoleru, G. Zhou, Q. Cao, P. Vicaire, J. A. Stankovic, T. F. Abdelzaher, J. Hui, and B. Krogh, "Vigilnet: An integrated sensor network system for energy-efficient surveillance," *ACM TOSN*, 2006.

[9] L. Luo, Q. Cao, C. Huang, L. Wang, T. F. Abdelzaher, J. A. Stankovic, and M. Ward, "Design, implementation, and evaluation of enviromic: A storage-centric audio sensor network," *ACM Trans. Sen. Netw.*, 2009.

[10] T. Zhu, Z. Zhong, Y. Gu, T. He, and Z.-L. Zhang, "Leakage-aware energy synchronization for wireless sensor networks," in *Proc. of the MobiSys*, 2009.

[11] S. Foster, "The energy efficiency of common household battery charging systems: results and implications," *Report on behalf of the Natural Resources Defense Council (NRDC)*, 2003.

[12] D. Panigrahi, S. Dey, R. Rao, K. Lahiri, C. Chiasserini, and A. Raghunathan, "Battery life estimation of mobile embedded systems," in *Proc. of the VLSID*, 2001.

[13] K. Lahiri, A. Raghunathan, S. Dey, and D. Panigrahi, "Battery-driven system design: A new frontier in low power design," in *Proc. of the VLSI Design/ASP-DAC*, 2002.

[14] T. Markel and A. Simpson, "Plug-in hybrid electric vehicle energy storage system design," in *Proc. of Advanced Automotive Battery Conference*, 2006.

[15] F. Zhang and Z. Shi, "Optimal and adaptive battery discharge strategies for cyber-physical systems," in *Proc. of CDC*, 2009.

[16] J. Flinn and M. Satyanarayanan, "Managing battery lifetime with energy-aware adaptation," *ACM Transactions on Computer Systems*, 2004.