

Energy-Efficient Multi-Hop Packet Transmission using Modulation Scaling in Wireless Sensor Networks

Yang Yu and Viktor K. Prasanna
 Department of EE-Systems
 University of Southern California
 Los Angeles, CA 90089-2562
 {yangyu, prasanna}@halcyon.usc.edu

Abstract— Wireless sensor networks are being developed for a variety of applications. Energy conservation for communication operations in such networks is crucial and challenging. This paper studies the problem of determining an energy-efficient schedule for packet transmission over a multi-hop communication path using modulation scaling. The goal is to minimize the maximal energy dissipation over all nodes while satisfying a specific latency constraint. A binary search based off-line algorithm is first developed. An on-line adaptive mechanism is then presented. Experimental results show that a significant amount of energy can be saved by using the off-line algorithm. Two runtime scenarios are used to demonstrate the ability of the on-line mechanism to adapt the modulation settings with respect to system variations in data packet size and latency constraint.

I. INTRODUCTION

Wireless sensor networks (WSN) are being developed for a wide range of civil and military applications, such as assembly line controlling, object tracking, infrastructure monitoring, habitat sensing, and battlefield surveillance [1], [2]. One basic feature of WSN applications is to send queries or gathered information in between the base station and the location(s) where the target phenomena is observed. Such feature usually requires multi-hop packet transmission if the distance from the base station to the phenomena is large. For example, a cluster-based sensor network [3], [4] necessitates a multiple-source single-recipient communication paradigm via multi-hop packet transmission (shown in Figure 1). Further, to reduce communication interference in highly dense wireless networks, the communication range of radios must be reduced. This results in multi-hop transmission as well.

A key concern in WSN is energy-efficiency. The large scale of such networks and the necessity of unattended operation over a long period of time requires careful management of the energy resources. In addition, wireless communication is a major source of power consumption during system operation – the ratio of the energy spent in sending one bit of information to the energy spent in executing one instruction is typically around several orders of magnitude [5]. It is therefore crucial to develop low-power radio modules (e.g., [6], [7]) as well

This work is supported by the DARPA Power Aware Computing and Communication Program under contract no. F33615-C-00-1633.

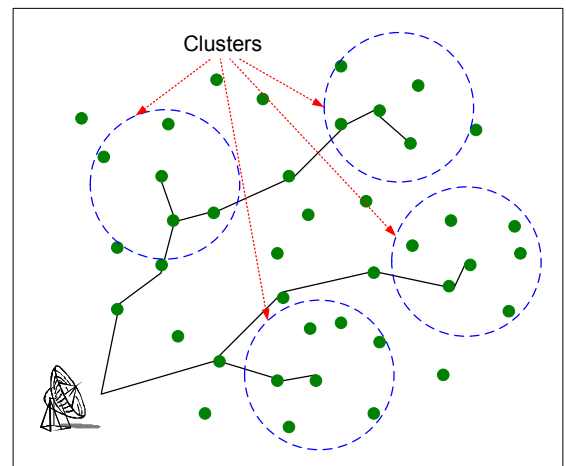


Fig. 1. Multi-hop packet transmission in a sensor network

as power-control schemes (e.g., [8]–[11] and energy-efficient packet scheduling mechanisms (e.g., [12]–[14]).

One useful technique for reducing the radio power consumption is the so-called modulation scaling [14], which explores the tradeoff between transmission energy and time duration by adapting the modulation level to match the traffic load. An important observation is that in many coding schemes, suppose $w(\tau)$ is the energy expended for transmitting a packet over τ units of time, $w(\tau)$ is a non-negative, monotonically decreasing, and strictly convex function of τ [13]. Several recent papers [12]–[14] have investigated techniques for power conservation in wireless communication. In [14], an on-line policy for a single-hop transmission link using modulation scaling is proposed. The problem of minimizing the energy dissipation for transmitting a set of packets over a single-hop link subject to a specific delay constraint is discussed in [13]. An extension of the problem that considers a single transmitter and multiple receivers is investigated in [12]. In both [13] and [12], the authors provide optimal off-line algorithms together with near-optimal on-line solutions.

While all the above efforts consider single-hop communication link, this paper studies the problem of scheduling packet transmission over a multi-hop communication path with the goal to achieve energy-efficiency. The transmission must be

completed within a specific end-to-end latency constraint. The forms of function $w(\tau)$ for different hops along the path can be different, depending on the radio component, distance, and environmental condition of each hop. One evaluation metric for energy-efficiency is the lifetime of the path, which is determined when the first sensor node along the path fails due to depleted energy. Thus, one intuitive objective is to minimize the maximal energy dissipation over all sensors along the path, with respect to their remaining energy.

The rest of the paper is organized as follows. The off-line scheduling problem for multi-hop packet transmission is described in Section II. A binary search based approximation algorithm for solving the off-line problem is presented. In Section III, an on-line mechanism for the scheduling problem is developed. Experimental results are shown in Section IV. Finally, the concluding remarks are given in Section V.

II. OFF-LINE SCHEDULING ALGORITHM FOR MULTI-HOP PACKET TRANSMISSION

A. Problem Definition

The underlying wireless network is modeled as a multi-hop communication path that involves n sensor nodes, denoted as $S_1 \rightarrow S_2 \rightarrow \dots \rightarrow S_n$, where \rightarrow represents a single-hop communication link, or simply hop. A packet of size s is needed to transmit from node S_1 to S_n in a given time interval (latency constraint) $[0, T]$. The value of T may be determined, for example, by the query rate at the base station.

It is well known that communication energy includes the energy for driving the transmit amplifier and the energy of transmit or receive electronic circuitry, such as filtering and modulating. For the sake of easy analysis, we focus on Quadrature Amplitude Modulation (QAM) scheme [15]. Note that the techniques presented in this paper are extendible to other modulation schemes as well. By fixing the symbol rate R_i for each hop, the time duration for transmission over $S_i \rightarrow S_{i+1}$, τ_i , can be modeled as:

$$\tau_i = \frac{s}{b_i \cdot R_i} \quad (1)$$

Further, the corresponding energy dissipation for sending the packet, E_i , can be modeled as [14]:

$$E_i = (P_i + D_i \cdot R_i) \cdot \tau_i = [C_i \cdot (2^{b_i} - 1) + D_i] \cdot \frac{s}{b_i} \quad (2)$$

where P_i is the transmission power; C_i is determined by the quality of transmission, in terms of Bit Error Rate, and the noise power; D_i is a device-dependent parameter that determines the power consumption of electronic circuitry; b_i is the constellation size in number of bits per symbol.

By substituting $b_i = \frac{s}{\tau_i \cdot R_i}$ in equation 2, it can be shown that E_i is a non-negative, strictly convex, and monotonically decreasing function of τ_i . Let $w_i(\tau_i)$ denote the function. With a set of parameter settings extracted from [16], $w_i(\tau_i)$ is illustratively depicted in Figure 2, where the domain of t_i is bounded by varying b_i from 2 to 8. A schedule of packet transmission, in terms of the vector $\vec{\tau} = \{\tau_i, i = 1, \dots, n-1\}$, is feasible if the transmission latency, $\sum_{i=1}^{n-1} \tau_i$, is within T .

Note the fact that the remaining energy of different nodes may not be identical, which needs to be considered for a

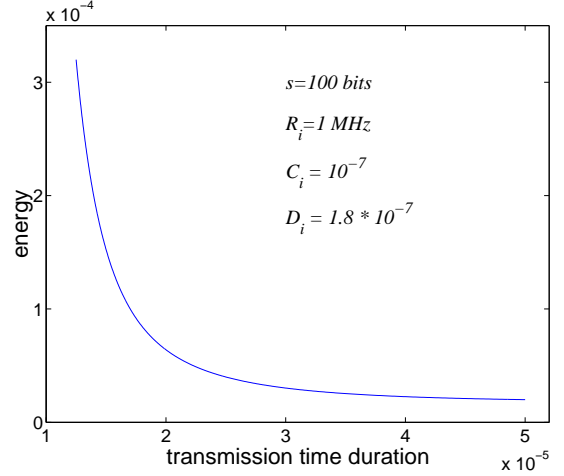


Fig. 2. Energy vs. transmission time with QAM scheme

balanced distribution of energy dissipation along the path. However, the above fact can be alternatively captured by normalizing parameters C_i and D_i with the remaining energy of node S_i . Thus, without loss of generality, each node is assumed to have one unit of remaining energy.

To emphasize on the main issues, the energy dissipation for receiving the data packet is not modeled. However, since energy dissipation for reception is fixed for each node, it can be easily incorporated into the above model and the algorithms presented in the paper. The off-line problem can now be formally stated as follows:

Given a series of consecutive single-hop communication links $S_1 \rightarrow S_2 \rightarrow \dots \rightarrow S_n$, and packet size s ;

find a feasible schedule of packet transmission, $\vec{\tau}$, so as to minimize the maximal energy dissipation over all nodes:

$$OBJ_{\vec{\tau}} = \max_{i=1}^{n-1} w_i(\tau_i) = \max_{i=1}^{n-1} [C_i \cdot (2^{\frac{s}{\tau_i R_i}} - 1) + D_i] \cdot \tau_i \cdot R_i$$

The node, S_i , that satisfies $w_i(\tau_i) = OBJ_{\vec{\tau}}$ is called the *critical node* for $\vec{\tau}$, or simply critical node. In general, critical node may not be unique.

B. Off-line Scheduling Algorithm

In this section, we describe the algorithm for the optimal off-line schedule. The necessary and sufficient conditions for optimality are first stated as follows.

Lemma 1: A schedule $\vec{\tau}$ is optimal if and only if $\vec{\tau}$ satisfies

$$(a) \quad \sum_i \tau_i = T; \text{ and}$$

$$(b) \quad w_1(\tau_1) = w_2(\tau_2) = \dots = w_{n-1}(\tau_{n-1}).$$

Proof: \Rightarrow The proof of condition (a) is trivial, since otherwise, we could simply increase the transmission time of all the critical nodes and reduce their energy dissipation.

To prove condition (b), let $\vec{\tau}$ be a feasible vector that does not satisfy condition (b). Further suppose that $\vec{\tau}$ is optimal, which means that $\vec{\tau}$ satisfies condition (a). Without loss of generality, we assume that the critical node for $\vec{\tau}$ is unique, denoted by S_i . Thus, for some j , $j \neq i$, we have $w_i(\tau_i) > w_j(\tau_j)$. Consider the schedule $\vec{\sigma}$ such that $\sigma_k = \tau_k$ for $k \neq$

$i, j, \sigma_i = \tau_i - \delta$, and $\sigma_j = \tau_j + \delta$, where $\delta > 0$ and $w_i(\sigma_i) \geq w_j(\sigma_j)$. It is easy to verify that $\vec{\sigma}$ is feasible and $OBJ_{\vec{\sigma}} < OBJ_{\vec{\tau}}$, which contradicts the optimality of $\vec{\tau}$ and completes the proof of necessity.

\Leftarrow Let $\vec{\tau}$ be a non-optimal schedule that satisfies both conditions (a) and (b). Further let $\vec{\sigma}$ be a feasible and optimal schedule. Obviously, we have $w_i(\sigma_i) < w_i(\tau_i)$ for $i = 1, \dots, n-1$. Due to the strictly non-decreasing property of $w_i(\cdot)$, we have $\sigma_i > \tau_i$. Hence $\sum_i \sigma_i > \sum_i \tau_i = T$, which contradicts the feasibility of σ and proves the lemma. \blacksquare

Though the optimal schedule can be obtained by solving the two set of equations in Lemma 1, it is quite challenging to directly solve the equations using algebraic techniques. However, Lemma 1 motivates a binary search based approximation algorithm to solve the problem.

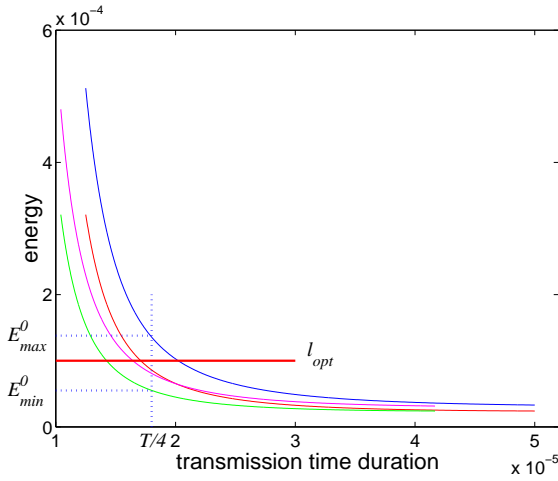


Fig. 3. Energy functions of a problem of n hops

For instance, consider an illustrative problem consisting of 4 hops (i.e., $n = 5$). The curves of the energy functions $w_1(\tau_1)$ to $w_4(\tau_4)$ are shown in Figure 3. Let $\vec{\tau}$ be the optimal schedule. From Lemma 1, the list of 4 points with coordinations $(\tau_i, w_i(\tau_i))$, $i = 1, \dots, 4$ must lie on a straight line that is parallel to the X axis. Let l_{opt} denote the line. The algorithm is performed in an iterative way, with τ_i^j denote the value of τ_i in the j -th iteration. Initially, we set $\tau_1^0 = \dots = \tau_4^0 = \frac{T}{4}$. If $w_1(\tau_1^0) = \dots = w_4(\tau_4^0)$ holds, we are done. Otherwise, let $E_{max}^0 = \max_i \{w_i(\tau_i^0)\}$ and $E_{min}^0 = \min_i \{w_i(\tau_i^0)\}$. Obviously, l_{opt} must lie in between lines $y = E_{min}^0$ and $y = E_{max}^0$. The exact position of l_{opt} can be found by using binary search. To terminate the search when a desired quality of the schedule is met, a user-specified parameter $\varepsilon \in (0, 1]$ is needed to indicate the tolerable difference between T and the total transmission delay of the schedule. The rationale behind it is that if the difference is small enough, the corresponding schedule is close to the optimal.

The pseudo code of the algorithm is shown in Figure 4. The value of b_i can be derived from equation 1 based on the final value of τ_i . Note that in practice, b_i can only take integer values. Thus, the final value of b_i is obtained after performing a ceiling function.

Begin

1. Set $j = 0$; $\tau_i^j = \frac{T}{n-1}$
2. Calculate E_{max}^j and E_{min}^j
3. **While** $\vec{\tau}^j$ is not optimal, **Do**
4. $j = j + 1$; $E_{mid} = \frac{E_{max}^j + E_{min}^j}{2}$
5. Calculate $\vec{\tau}^j$ by solving equations $w_i(\tau_i^j) = E_{mid}$
6. **If** $\sum_i \tau_i^j \in [T(1 - \varepsilon), T]$, $\vec{\tau}^j$ is optimal
7. **Else If** $\sum_i \tau_i^j < T$
8. $E_{max}^j = E_{max}^{j-1}$; $E_{min}^j = E_{mid}$
9. **Else**
10. $E_{max}^j = E_{mid}$; $E_{min}^j = E_{min}^{j-1}$

End

Fig. 4. Pseudo code for the off-line algorithm

It is easy to verify that the time complexity of the algorithm is $n\alpha \log(E_{max}^0 - E_{min}^0)$, where α is the time used to solve one equation of form $w_i(\tau_i) = E_{opt}$ in line 5. An option to help reduce the computation requirement for solving the equation is through a pre-computed lookup-table, where interpolation for approximated results may be used to reduce the size of the table. Note that our problem can also be solved by general convex optimization methods. However, without exploiting the special structure of the problem, general methods need to perform a significant amount of computation.

III. ON-LINE SCHEDULING

In this section, we describe a simple, adaptive on-line scheduling mechanism. First, we discuss the assumptions needed for implementing the mechanism.

It is assumed that (1) for each transmitted packet, node S_n knows the total time duration of the transmission; and (2) each node S_i is aware of the value of E_i . Assumption (1) can be achieved by synchronizing nodes S_1 and S_n . To justify assumption (2), consider equation 2, D_i is device-dependent and can be obtained from the radio vendor, while P_i , s , b_i , and R_i are system-aware parameters.

The on-line scheduling mechanism works as follows. The initial value of b_i is set to the highest level for each node. Before the packet is transmitted over a hop, the sender of the hop, say S_i , calculates and attaches the value of E_i onto the packet. Node S_n is responsible for extracting the information of E_i for all nodes after receiving the packet. After each user-specified time window, node S_n checks if the average transmission latency in the last time window is relatively small compared with the latency constraint, T , – for example, the latency is small if it is less than 95% of T . If yes, S_n sends a feedback packet to S_1 by reversely traversing the path. The feedback packet contains three fields: *target* - the index of the critical node, E_{next} - the second largest value of E_i , and Δ - the difference between T and the average transmission latency in the last time window. The value of b_i of all nodes are fixed during each time window. Upon the reception of a feedback packet, each node, S_i , checks if it is the node indicated by the

target field in the packet. If yes, the value of b_i is decreased till the corresponding value of E_i is less than E_{next} , while ensuring that the incurred increase in τ_i is no larger than Δ . Recall that b_i can only take positive integers in practice; thus, the above procedure stops when $b_i = 1$ for all nodes along the path.

Variations in the system or environmental conditions may cause violation of the latency constraint during packet transmission. Examples of such variations include changes in data size or latency constraint. Thus, the on-line mechanism must be capable of adapting the modulation settings when latency violation is detected. One method is that upon the reception of a feedback packet with a negative value of Δ , each node, S_i , restores the value of b_i to the highest level. However, this method may lead to unnecessarily high energy dissipation, if the variation in data size or latency constraint is not dramatical. Thus, a more efficient method is to restore the value of b_i to $\lceil \frac{b_i}{\beta} \rceil$, where $\beta \in (0, 1)$ is a user-specified parameter and controls the speed for increasing b_i . Note that other methods are possible to further explore the tradeoff between the energy conservation and the quickness of the system response.

IV. EXPERIMENTAL RESULTS

A simulator for evaluating the scheduling mechanisms described in this paper was developed using the PARSEC [17] software, which is a discrete-event simulation language. The simulations were conducted in two steps, for evaluating the off-line and on-line mechanisms, respectively.

A. Off-line Performance

Experimental Setup: The size of the packet, s , was set to 400 bits. The length of the path, $n - 1$, was selected from the set $\{5, 10, 20\}$. For each value of n , a set of experiments were conducted with parameters C_i , D_i , and R_i modeled as random variables with uniform distribution between $(1 \pm 0.2)10^{-7}$, $(2 \pm 0.2)10^{-7}$, and $(1 \pm 0.5)10^6$, respectively. The above numbers are based on the data presented in [16]. The range of b_i was set to $[2, 12]$. The value of ε was set to 0.05.

For each packet transmission, the shortest time to complete can be calculated as if the modulation settings of all the nodes are set to the highest levels. Let λ be the ratio of the shortest transmission time over the end-to-end latency constraint. Ratio λ indicates the stringency of the latency constraint and hence, the space for conserving energy at the cost of increased latency. In our experiments, the value of λ was varied from 0 to 1 with an increment of 0.05. Intuitively, when λ is set to one, zero percent of energy conservation can be achieved. On the other hand, when λ is zero, no energy is dissipated for the transmission since the time duration can be infinity.

Energy Conservation vs. λ : The results of the energy conservation achieved by using our off-line algorithm with different settings of λ is shown in Figure 5. Due to the space limitation, only the result for $n - 1 = 10$ is shown. However, very similar results were observed for cases when $n - 1 = 5$ or 20. Let E_{raw} denote the maximal energy dissipation over all nodes without using modulation scaling, (i.e., every node sends data using the highest speed). The energy conservation is then

defined as $\frac{E_{raw} - OBJ_{\vec{\tau}}}{E_{raw}}$, where $\vec{\tau}$ is the schedule obtained by the off-line algorithm.

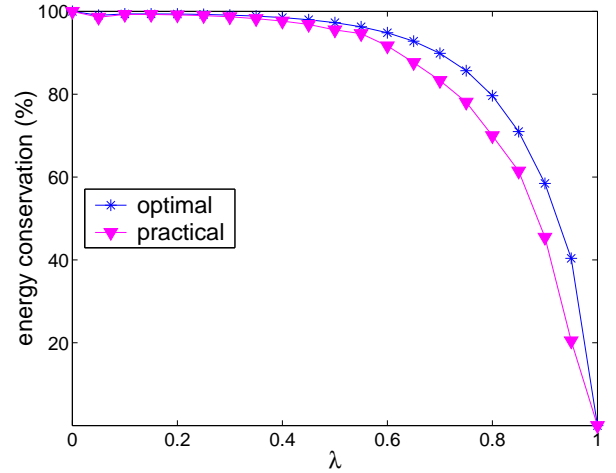


Fig. 5. Energy conservation vs. λ

The plotted data was averaged over more than 100 iterations so that each data point has a 95% confidence interval with a 5% (or better) precision. The optimal curve is obtained using our off-line algorithm when b_i can be a real number between $[2, 12]$, while the practical curve is obtained after performing a ceiling function for the optimal b_i 's. A slight degradation of performance can be observed between the practical and optimal curve. More importantly, the plot clearly shows that the energy conservation roughly follows a concave function of λ , which means considerable energy gain can still be achieved when the latency constraint is stringent. For example, when $\lambda = 0.5$, more than 95% of the energy can be conserved. When $\lambda = 0.8$, the practical energy conservation still reaches 70%. The above property reveals the significance of modulation scaling in the sense that even a small latency laxity can be used to trade for a large amount of energy conservation.

B. On-line Behavior

Experimental Setup: To study the behavior of the on-line mechanism with respect to variations in packet size and latency constraint, the values of s and λ were dynamically varied during runtime. Two scenarios of variations, denoted A and B, were investigated. The variations in environmental condition, such as noise power, were not simulated. Nevertheless, such variations can be incorporated into our simulator.

The settings of C_i , D_i , R_i , d_i , and ε were the same as the settings in Section IV-A. The number of hops was set to 10. The size of the time window was set to 0.0008 second.

Scenario A: In scenario A, s was fixed at 400, while λ was set to 0.8, 0.6, 0.5, 0.7, and 0.8 at time 0, 0.1, 0.15, 0.3, and 0.4 second, respectively. The runtime behavior of the system is depicted in Figure 6. The time instances of the above variations in λ are indicated by the arrows in the figure.

It can be observed from Figure 6(a) that when λ is fixed, the actual transmission latency gradually increases till close to the latency constraint as the time advances. At time 0.3 second, λ is varied from 0.5 to 0.7, which causes a violation

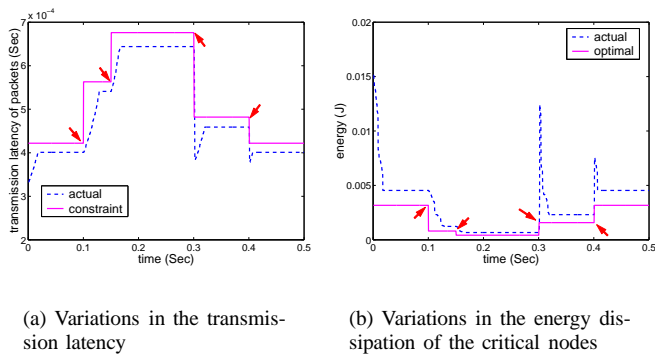


Fig. 6. Runtime behavior of the system in Scenario A

of the latency constraint. Due to the feedback mechanism, the transmission latency dramatically decreases as the modulation settings of nodes are restored to higher levels.

In addition, it can be observed from Figure 6(b) that the energy dissipation of critical nodes decreases till close to the optimal value as the time advances. At time 0.3 second, the energy dissipation dramatically increases as the modulation settings of nodes are restored to higher levels. After that, the energy dissipation drops again as the time advances.

Scenario B: In scenario B, the latency constraint was fixed at $4.2E-4$ second, while s was set to 400, 350, 300, 250, and 400 at time 0, 0.1, 0.15, 0.3, and 0.4 second, respectively. The runtime behavior of the system is depicted in Figure 7. The time instances of the above variations in s are indicated by the arrows. An analysis similar to the one in Scenario A can be performed based on the results shown in Figure 7. In short, our on-line mechanism is capable of effectively adapting the modulation levels of nodes with respect to various system variations, such as data size and latency constraint.

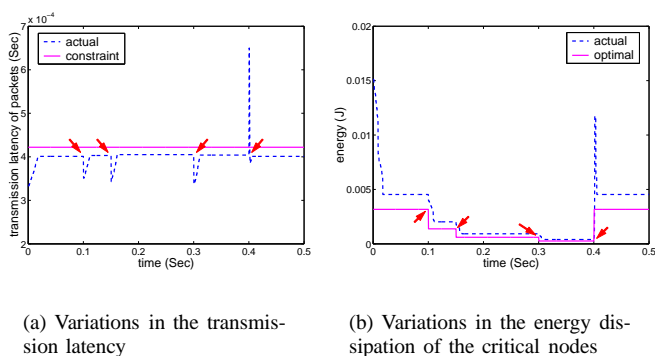


Fig. 7. Runtime behavior of the system in Scenario B

V. CONCLUDING REMARKS

This paper studied the problem of finding an energy-efficient schedule for packet transmission over a multi-hop communication path using modulation scaling. An off-line approximation algorithm was first developed, followed by an on-line mechanism. Simulation results show that the off-line algorithm can save significant energy at the cost of increased

transmission latency without violating the latency constraint. In addition, two runtime scenarios demonstrate the ability of our on-line mechanism to adapt the modulation levels with respect to various system variations.

Several extensions of the problem studied in this paper are worth further investigating. From a protocol perspective, variations of the proposed on-line mechanism can be developed for improved robustness and performance. From network topology perspective, one extension is to schedule a packet over a more general network connection, such as a spanning tree. This models a communication pattern similar to multicast. Another extension is to simultaneously schedule multiple packets over an ad hoc network structure – probably using the multi-source single-recipient paradigm as for sensor networks. In this context, several issues need to be considered, including contention avoidance in the network and performance tradeoff between different packets.

REFERENCES

- [1] D. Estrin, L. Girod, G. Pottie, and M. Srivastava, "Instrumenting the world with wireless sensor networks," in *The International Conference on Acoustics, Speech and Signal Processing*, May 2001.
- [2] G. J. Pottie and W. J. Kaiser, "Wireless integrated network sensors," *Communications of the ACM*, vol. 43, no. 5, pp. 551–558, May 2000.
- [3] M. Singh and V. K. Prasanna, "A hierarchical model for distributed collaborative computation in wireless sensor networks," to appear on 5th Workshop on Advances in Parallel and Distributed Computational Models.
- [4] Y. Yu, B. Krishnamachari, and V. K. Prasanna, "Cluster-based lightweight middleware for a class of wireless sensor networks," submitted to *IEEE Network Magazine*.
- [5] M. Singh and V. K. Prasanna, "System level energy tradeoffs for collaborative computation in wireless networks," in *IEEE IMPACCT Workshop*, May 2002.
- [6] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, and K. Pister, "System architecture directions for networked sensors," in *9th International Conference on Architectural Support for Programming Languages and Operating Systems*, 2000.
- [7] J. Rabaey, J. Ammer, T. Karalar, S. Li, B. Otis, M. Sheets, and T. Tuan, "Picoradios for wireless sensor networks: The next challenge in ultra-low-power design," in *International Solid-State Circuits Conference*, Feb. 2002.
- [8] E. Armanian, D. Falconer, and H. Yanikomeroglu, "Adaptive modulation, adaptive coding, and power control for fixed cellular broadband wireless systems," Submitted to *IEEE WCNC 2003*.
- [9] A. F. Almutairi, S. L. Miller, H. A. Latchman, and T. F. Wong, "Power control algorithm for mmse receiver based cdma systems," *IEEE Communications Letters*, vol. 4, no. 11, pp. 346–348, Nov. 2000.
- [10] S.-L. Kim, Z. Rosberg, and J. Zander, "Combined power control and transmission rate selection in cellular networks," in *IEEE VTC*, 1999.
- [11] K. k. Leung and L.-C. Wang, "Integrated link adaptation and power control for wireless IP networks," in *IEEE VTC*, 2000, pp. 2086–2092.
- [12] A. E. Gamal, C. Nair, B. Prabhakar, E. Uysal-Biyikoglu, and S. Zahedi, "Energy-efficient scheduling of packet transmissions over wireless networks," in *INFOCOM*, 2002.
- [13] B. Prabhakar, E. Uysal-Biyikoglu, and A. E. Gamal, "Energy-efficient transmission over a wireless link via lazy packet scheduling," in *INFOCOM*, 2001.
- [14] C. Schurgers, O. Aberhorne, and M. Srivastava, "Modulation scaling for energy-aware communication systems," in *ISLPED*, 2001, pp. 96–99.
- [15] T. Ue, S. Sampei, N. Morinaga, and K. Hamaguchi, "Symbol rate and modulation level-controlled adaptive modulation/TDMA/TDD system for high-bit rate wireless data transmission," *Trans. on Vehicular Technology*, vol. 47, no. 4, pp. 1134–1147, Nov. 1998.
- [16] K. Cho and H. Samuelli, "A 8.75Mbaud single-chip digital QAM modulator with frequency-agility and beamforming diversity," in *IEEE 2000 Custom Integrated Circuits Conference*, May 2000.
- [17] PARSEC Project. [Online]. Available: <http://pcl.cs.ucla.edu/projects/parsec>