



Optimal on-off control for a class of discrete event systems with real-time constraints

Lei Miao¹ · Lijian Xu² · Dingde Jiang³

Received: 6 September 2018 / Accepted: 14 January 2019 / Published online: 26 January 2019
© Springer Science+Business Media, LLC, part of Springer Nature 2019

Abstract

This paper studies an optimal ON-OFF control problem for a class of discrete event systems with real-time constraints. Our goal is to minimize the overall costs, including the operating cost and the wake-up cost, while still guaranteeing the deadline of each individual task. In particular, we consider the homogeneous case in which it takes the same amount of time to serve each task and each task needs to be served by d seconds upon arrival. The problem involves two subproblems: (i) finding the best time to wake up the system and (ii) finding the best time to let the system go to sleep. We study the two subproblems in both off-line and on-line settings. In the off-line case that all task information is known a priori, we combine sample path analysis and dynamic programming to come up with the optimal solution. In the on-line scenario where future task information is completely unknown, we show that the optimal time to wake up the system can be obtained without relying on future task arrivals. We also perform competitive analysis for on-line control and derive the competitive ratios for both deterministic and random controllers.

Keywords Discrete event systems · Real-time systems · Quality-of-service · Optimization · Dynamic programming · Competitive ratio

The authors' work is supported in part by a start-up funding provided by Middle Tennessee State University.

✉ Lei Miao
lei.miao@mtsu.edu

Lijian Xu
xul@farmingdale.edu

Dingde Jiang
jiangdd@uestc.edu.cn

¹ Mechatronics Engineering, Middle Tennessee State University, 1301 East Main Street, Box 19, Murfreesboro, TN 37132-0001, USA

² Department of Electrical and Computer Engineering Technology, Farmingdale State College, Farmingdale, NY 11735, USA

³ School of Astronautics and Aeronautic, University of Electronic Science and Technology of China, Chengdu, 611731, China

1 Introduction

There exists a large amount of Discrete Event Systems (DESs) that involve allocation of resources to satisfy real-time constraints. One commonality of these DESs is that certain tasks must be completed by their deadlines in order to guarantee Quality-of-Service (QoS). Examples arise in wireless networks and computing systems, where communication and computing tasks must be transmitted/processed before the information they contain becomes obsolete (Miao et al. 2017; Liu 2000), and in manufacturing systems, where manufacturing tasks must be completed before the specified time in the production schedule (Pepyne and Cassandras 2000). Another commonality of these DESs is that they all require the minimization of cost (e.g., energy). An interesting question then arises naturally: *how can we allocate resources to such DESs so that the cost is minimized and the real-time constraints are also satisfied?* To answer this question, one often has to study the trade-off between minimizing the cost and satisfying the real-time constraints: processing the tasks at a higher speed makes it easier to satisfy the real-time constraints but harder to reduce the cost; conversely, processing the tasks at a lower speed makes it harder to satisfy the real-time constraints but easier to reduce the cost. This trade-off is often referred to as the energy-latency trade-off and has been widely studied in the literature (Miao et al. 2017; Gamal et al. 2002; Zafer and Modiano 2009a).

In this paper, our objective is to utilize the energy-latency trade-off to minimize the cost while guaranteeing the real-time constraint for each task. Different from most existing papers that assume the system's service rate (the control variable) is a continuous function of time, we assume that the DES only operates at one of the two states: ON and OFF. One motivating example of such DES is wireless sensor networks, in which operation simplicity must be maintained. For example, the radio of a ZigBee wireless device can be either completely off or transmitting at a fixed-rate, e.g., 250kb/s in the 2.4GHz band. Another difference between this paper and others is that we assume that a wake-up cost is incurred whenever the system transits from the OFF state to the ON state.

We solve both off-line and on-line optimal ON-OFF control problems. Our main contributions are two-fold: (i) We combine *sample path analysis* and *Dynamic Programming* (DP) to obtain the optimal off-line solution and (ii) We perform competitive analysis and derive the competitive ratios of both deterministic and random on-line controllers. Some results of this paper are previously shown in two conference papers: Miao and Xu (2015) and (Miao 2017), which primarily focus on off-line control. One new contribution of this paper is the competitive analysis for on-line control. Another new contribution is that we introduce an idling cost in the system model. We point out that the addition of this idling cost makes our problem formulation more realistic because it often exists in real-world applications. For example, energy is consumed when a motor is spinning without any load attached and when a sensor is turned on, but not actively processing information.

Due to page limitation, we do not include the results and discussions that are available in the conference papers and all the proofs; instead, we let the readers refer to Miao et al. (2018) for all these details. The organization of the rest of the paper is as follows: in Section 2, we discuss related work; we introduce the system model and formulate our optimization problem in Section 3; the off-line and on-line results are presented in Sections 4 and 5, respectively; finally, we conclude in Section 6.

2 Related work

There are two lines of work that are closely related to this paper. One is transmission scheduling for wireless networks, in which the transmission rate of a wireless device is adjusted so as to minimize the transmission cost and satisfy real-time constraints. This line of work is initially studied in Uysal-Biyikoglu et al. (2002) with follow-up work in Gamal et al. (2002) where a homogeneous case is considered for a convex optimization problem, assuming all packets have the same deadline and number of bits. The rate of convergence of the proposed “MoveRight” algorithm is obtainable for a special case of the problem when all packets have identical energy functions. Zafer and Modiano (2009b) study an optimal rate control problem over a time-varying wireless channel, in which the channel state was modeled as a Markov process. In particular, they consider the scenario that B units of data must be transmitted by a common deadline T , and they obtain an optimal rate-control policy that minimizes the total energy expenditure subject to short-term average power constraints. In Zafer and Modiano (2007, 2008), the case of identical arrival time and individual deadline is studied by Zafer et. al. In Chen et al. (2007), the case of identical packet size and identical delay constraint is studied by Neely et. al. They extend the result for the case of individual packet size and identical delay constraint in Chen et al. (2009). In Zafer and Modiano (2009a), Zafer et. al. use a graphical approach to analyze the case that each packet has its own arrival time and deadline. However, there are certain restrictions in their setting; for example, the packet that arrives later must have later deadlines. Wang and Li (2013) analyze scheduling problems for bursty packets with strict deadlines over a single time-varying wireless channel. Assuming slotted transmission and changeable packet transmission order, they are able to exploit structural properties of the problem to come up with an algorithm that solves the off-line problem. In Poulakis et al. (2013), Poulakis et. al. also study energy efficient scheduling problems for a single time-varying wireless channel. They consider a finite-horizon problem where each packet must be transmitted before D_{\max} . Optimal stopping theory is used to find the optimal start transmission time between $[0, D_{\max}]$ so as to minimize the expected energy consumption and the average energy consumption per unit of time. Zhong and Xu (2008) formulated optimization problems that minimize the energy consumption of a set of tasks with task-dependent energy functions and packet lengths. In their problem formulation, the energy functions include both transmission energy and circuit power consumption. To obtain the optimal solution for the off-line case with backlogged tasks only, they develop an iterative algorithm RADB whose complexity is $O(n^2)$ (n is the number of tasks). The authors show via simulation that the RADB algorithm achieves good performance when used in on-line scheduling. Miao et al. (2017) studies a transmission control problem for task-dependent cost functions and arbitrary task arrival time, deadline, and number of bits. They propose a GCTDA algorithm that solves the off-line problem efficiently by identifying certain critical tasks. The GCTDA algorithm is an extension to the CTDA algorithm (Mao et al. 2007) designed by Mao and Cassandras for dynamic voltage scaling related applications. They extend the CTDA algorithm to multilayer scenarios in Mao and Cassandras (2014). Our model is different from all the above works by letting the system operate in one of the discrete modes and also including a wake-up cost at each time instant that the system transitions from OFF to ON state.

The other line of research studies On-OFF scheduling in Wireless Sensor Networks (WSNs). Solutions in the Medium Access Control (MAC) layer, such as the S-MAC protocol (Ye et al. 2004), have been developed to coordinate neighboring sensors’ ON-OFF schedule in order to reduce both energy consumption and packet delay. These approaches

do not provide specific end-to-end latency guarantee. In Lai and Paschalidis (2006), routing problems are considered in WSNs where each sensor switches between ON and OFF states. The authors formulate an optimization problem to pick the best path that minimizes the weighted sum of the expected energy cost and the exponent of the latency probability. In another work in Ning and Cassandras (2006), Ning and Cassandras formulate a dynamic sleep control problem in order to reduce the energy consumed in listening to an idle channel. The idea is to sample the channel more frequently when it is likely to have traffic and less frequently when it is not. The authors extend their work in Ning and Cassandras (2008), by formulating an optimization problem with the goal of minimizing the expected total energy consumption at the transmitter and the receiver. Dynamic programming is used to come up with an optimal policy that is shown to be more effective in cost saving than the fixed sleep time. Cohen and Kapchits (2009) studies the ON-OFF scheduling in wireless mesh networks. By assuming a fixed routing tree topology used for task transmission, each child in the tree knows exactly when its parents will wake up, and the traffic is only generated by the leaves of the tree, the authors formulate and solve an optimization problem that minimizes the total transmission energy cost while satisfying the latency and maximum energy constraints on each individual node. The major difference between this paper and the existing ones in this line of research is that we study a system with a real-time constraint for each individual task. To the best of our knowledge, ON-OFF scheduling with a real-time constraint for each individual task has not been studied extensively.

It is worth noting that there also exists papers related to the service rate control problem, in which the optimal service rate policy of either single-server or multi-server queueing systems are derived in order to minimize an average cost. A recent representative work along this line can be found in Xia et al. (2017) where Xia et al. study the service rate problem for tandem queues with power constraints. They formulate the model as a Markov decision process with constrained action space and use sensitivity-based optimization techniques to derive the conditions of optimal service rates, the optimality of the vertexes of the feasible domain for linear and concave operating cost, and an iterative algorithm that provides the optimal solution. Our problem formulation is different from these works in two aspects: (i) We consider tasks with real-time constraints and (ii) We include system wake-up cost on top of the service cost.

3 System model and problem formulation

We consider a finite horizon scenario that a DES processes N tasks with real-time constraints. In particular, task i , $i = 1, \dots, N$, has arrival time a_i (generally random), deadline $d_i = a_i + d$, and B number of operations. Both d and B are constants. In the *off-line* setting, we assume that the task arrival time a_i is known to the controller a priori. The DES can only operate in one of the two modes: ON and OFF. When it is in the OFF mode, there is no operating cost associated. When it is in the ON or active mode, the system can either be *busy* or *idling*. When the system is busy, it processes the tasks at a constant rate R with fixed operating cost C_B per unit time. When the system is idling, no tasks are waiting to be served, and the system cost is C_I ($C_I \leq C_B$) per unit time. Furthermore, we assume that whenever a transition from the OFF mode to the ON mode occurs, a fixed wake-up cost C_W is incurred; examples of such costs include: the large amount of current (known as inrush current) required when a motor is turned on, the energy needed to initialize electric circuits when RF radio is turned on in a wireless device, and so on. Note that the wake-up

cost may also include system wearout cost, if the system can only be turned on for certain number of times during its lifetime. In our previous work in Miao and Xu (2015) and Miao (2017), $C_I = C_B$. As we will show later, when C_I is different from C_B , it does not make the analysis significantly harder, and the off-line optimal solution can still be obtained by DP.

As we mentioned earlier, the task information is known to the controller a priori in the off-line setting. Our objective is to find the optimal ON and OFF time periods so as to (i) finish all the tasks by their deadlines and (ii) minimize the cost.

Definition 1 Suppose the system is woken up at t_1 , put to sleep at t_2 ($t_1 < t_2$), and kept active from t_1 to t_2 . Then, we call the time interval $[t_1, t_2]$ an **Active Period (AP)**.

Definition 2 In any AP, the periods during which the system is actively serving tasks are known as **Busy Periods (BPs)**. The rest of the time periods in that AP are known as **Idle Periods (IPs)**.

Let $r(t)$ be the rate that the system is capable of serving tasks at time t . It is piecewise constant and at any given time t , it can only be either 0 (when the system is OFF) or R (when the system is ON). Note that $r(t)$ is the potential service rate, instead of the the actual service rate, since the system is only serving tasks during the **BPs**, not the **IPs**.

We now introduce the control variables. Our first control variable is α , the number of APs. The second control variable is a $\alpha \times 2$ array \mathbf{t} that contains 2α time instants. These time instants satisfy: $t_{i,1} < t_{i,2} < t_{j,1} < t_{j,2}, \forall i, j \in \{1, \dots, \alpha\}, i < j$ and define α number of APs. The off-line problem $Q(1, N)$ can then be formulated:

$$\begin{aligned} \min_{\alpha, \mathbf{t}} J &= \alpha C_W + \sum_{i=1}^{\alpha} [C_I(t_{i,2} - t_{i,1} - \tau_{i,B}) + C_B \tau_{i,B}] \\ \text{s.t. } &\int_{\max(a_j, x_{j-1})}^{x_j} r(t) dt = B, \\ &x_j \leq d_j, x_0 = 0, j = 1, \dots, N \\ &r(t) = R \sum_{i=1}^{\alpha} [u(t - t_{i,1}) - u(t - t_{i,2})] \end{aligned}$$

where x_j is the departure time of task j , $u(t)$ is the unit step function, and $\tau_{i,B}$ is the length of the busy periods in the i -th AP. The first constraint ensures that exactly B number of operations are executed for each task. The second one is the real-time constraint. The third one makes sure that the processing rate is R only during each AP. Note that $\tau_{i,B}$ is dependent on the number of tasks served in AP_i . To represent $\tau_{i,B}$, we use N_i^S and N_i^E to denote the first (starting) task and the last (ending) task in AP_i , respectively: $N_i^E = \arg \max_{j \in \{1, \dots, N\}} (d_j \leq$

$t_{i,2})$, $N_i^S = \arg \min_{j \in \{1, \dots, N\}} (a_j \geq t_{i,1})$, and $\tau_{i,B} = \max((N_i^E - N_i^S + 1) \frac{B}{R}, 0)$.

Notice that $Q(1, N)$ above may not always be feasible. In this paper, we only consider the case that $Q(1, N)$ is indeed feasible. See Assumption 1 and Lemma 1 in Miao et al. (2018) for the assumption that guarantees feasibility.

$Q(1, N)$ is a hard optimization problem, due to the nondifferentiable terms in the constraints and the objective function. It cannot be easily solved by standard optimization software. In what follows, we will first discuss optimal off-line control, using which we will then establish the results for on-line control.

4 Off-line control

In this section, we focus on the off-line control problem, in which all task arrivals are known to us a priori. We need to find out when the system should wake up and start to serve the first task in an AP. Similar to the “just-in-time” idea exploited in Gamal et al. (2002) for adaptive modulation, the system should wake up as late as possible so that it may potentially reduce the active time. The question is how late the system should wake up. This is answered by Lemmas 2-4 in Miao et al. (2018).

We now find out when the system should go to sleep. Lemma 5 in Miao et al. (2018) provides a sufficient but not necessary condition of ending an AP on the optimal sample path. Let $d_0 = -\infty$ and $a_{N+1} = \infty$. We introduce the following definition.

Definition 3 Consecutive tasks $\{k, \dots, n\}$, $1 \leq k \leq n \leq N$, belong to a *super active period* (SAP) in problem $Q(1, N)$ if $d_{k-1} + C_W/C_I < a_k$, $d_n + C_W/C_I < a_{n+1}$, and $d_j + C_W/C_I \geq a_{j+1}$, $\forall j \in \{k+1, \dots, n-1\}$.

Each SAP contains one or more APs. SAPs can be easily identified by simply examining all the task deadlines and arrival times and applying Lemma 5 in Miao et al. (2018). It implies that instead of working on the original problem $Q(1, N)$, we now only need to focus on each SAP, which is essentially a subproblem $Q(k, n)$.

We now define our decision points in each SAP. A decision point x_t , $t \in \{k, \dots, n-1\}$, is the departure time of task t satisfies $x_t < a_{t+1}$. If $x_t \geq a_{t+1}$, 0 then x_t is not a decision point because the system should stay active at x_t and process task $t+1$. At each decision point, the control is letting the system either go to sleep or stay awake, and the optimal control depends on future task arrivals. A first look at the problem seems to suggest that the problem is intractable; however, a closer look indicates that the off-line optimal ON-OFF control problem can be solved by DP, which has been widely used to solve a large class of problems with special structural properties. In the context of DES, however, its usage has been very limited to date. For example, in Mao et al. (2007) and Miao et al. (2017) where the problem formulation is similar to the one in this paper, both CTDA and GCTDA algorithms are not DP-based. We will show next that for the DES studied by this paper, DP and sample path analysis can be used together to obtain the optimal solution. In particular, it is done by introducing two types of tasks: starting and following.

Definition 4 In problem $Q(k, n)$, where tasks $\{k, \dots, n\}$ form an SAP, the first task of any AP is called a *starting* task. Tasks that are not starting tasks are known as *following* tasks.

For any task $i \in \{k, \dots, n\}$, it must either be a starting task or a following one. We use $Q^S(i, n)$ and $Q^F(i, n)$ to denote the optimization problems of serving tasks $\{i, \dots, n\}$ when task i is a starting and following task, respective. Let J_i^S and J_i^F be the minimum cost of $Q^S(i, n)$ and $Q^F(i, n)$, respectively.

Theorem 1 J_k^S is the optimal cost of problem $Q(k, n)$.

This theorem shows that when the algorithm in Table 1 in Miao et al. (2018) stops, J_k^S is the optimal cost of problem $Q(k, n)$. The corresponding optimal control, i.e., the starting time and ending time of each AP, can be traced back iteratively by identifying the J_i^S or J_i^F that each J_{i-1}^S or J_{i-1}^F points to. The procedure is provided in Table 4 in Miao et al. (2018).

Next, we use simulation results to show how the optimal solution performs compared with a naive approach, in which the controller simply goes to sleep when there is no backlog and wakes up when a new task arrives. Let optimal to naive ratio be the ratio between the optimal cost and the cost of the naive controller. Figure 1 shows how the optimal to naive ratio varies when the task arrival process and the wake-up cost C_W change. In the simulation, we have 100 runs that correspond to 100 maximum interarrival time from $1ms$ to $100ms$ with step size $1ms$. For the purpose of statistical regularity, each run has 1000 tasks. The interarrival time between two adjacent tasks is uniformly distributed between 0 and the maximum interarrival time in each run. The values of the other parameters are as follows: $d = 20ms$, $C_B=30mW$, $C_I=100\mu W$, and $\beta = 1ms$.

We have a couple of observations. First, the cost saving of the optimal solution is greater when C_W is larger. Second, the maximum cost saving occurs when the interarrival time is not too small or too large: when it is too small, a single AP will be sufficient to complete all the tasks, and the optimal and the naive solutions are essentially the same; when it is too large, many APs are needed, and the advantage of the optimal controller gets smaller. As we can see from the result, the cost saving of the DP algorithm in the $C_W = 28mJ$ case is as large as 50%, and it will be ever greater when C_W is higher.

5 On-line control

In this section, we study on-line control where future task arrival information is unknown to the controller. Essentially, the controller needs to decide the starting time and ending time for each AP. When to start an AP has been previously studied and see Lemma 6 in Miao et al. (2018) for details. We now focus on ending an AP in on-line control.

When all backlogged tasks have been served in an on-line setting, the controller needs to decide when to end an AP and put the system to sleep. This decision depends on future task information and the values of idling cost C_I and wake-up cost C_W . For example, if the next task $t + 1$ arrives very soon, the optimal control at decision point x_t might be letting

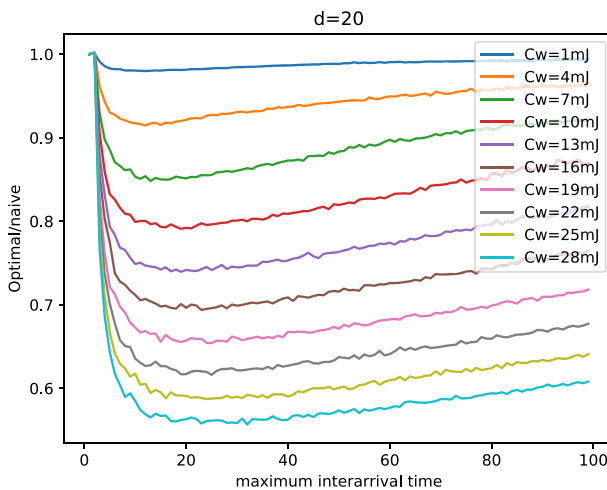


Fig. 1 Optimal to naive ratio under various wake-up cost and interarrival time

the system stay active; conversely, if the next task $t + 1$ arrives after a long time, then the system perhaps should go to sleep at decision point x_t .

When some future task information is known, techniques such as Receding Horizon Control (RHC) can be utilized to make decisions. In this paper, we focus on the scenario that no future task information is available at all.

In general, the control at each decision point is the following: let the system stay awake for another θ_t seconds. If no task arrives within the θ_t seconds, then put the system to sleep after θ_t seconds; o.w., serve the newly arrived tasks and wait for the next decision point. Note that the subscript t indicates θ_t could be different at each decision point.

Let J^* be the optimal cost of the off-line problem $Q(1, N)$ and \tilde{J} be the cost of the on-line controller. Our objective is to develop competitive on-line controllers which can quantify their worst-case performance deviation from the optimal off-line solution.

One challenge of competitive analysis is to find out the worst-case scenario. In our problem, the unnecessary cost in on-line control occurs when the system is idling: the controller must decide if and when to sleep. Therefore, the worst case occurs when each AP contains only one task so that the decision has to be made over and over again for every single task. This property actually simplifies our analysis, and in particular, we tackle the competitive ratio problem from two different aspects: a deterministic controller and a randomized one.

5.1 Deterministic controller

We first consider a deterministic controller in which θ_t is a fixed constant value θ . The on-line controller is c -competitive if $\tilde{J}(I, \theta) \leq cJ^*$, $\forall I \in \mathcal{I}$, where \mathcal{I} is the set of all possible task arrival instances and I is one task arrival instance. c is called the competitive ratio of the deterministic on-line controller and is essentially the *upper-bound* (i.e., *worst case*) of the ratio between the on-line cost \tilde{J} and the off-line optimal cost J^* .

Lemma 1 *The best competitive ratio c^* of a deterministic controller is obtained when $\theta = C_W/C_I$, and $\lim_{N \rightarrow \infty} c^* = (2 + \gamma)/(1 + \gamma)$, where N is the number of tasks and $\gamma = C_B\beta/C_W$.*

Lemma 1 shows that the competition ratio of a deterministic algorithm depends on the ratio between $C_B\beta$, the cost of serving one task, and C_W , the cost of waking up the system. If this ratio is very small, then the competitive ratio is close to 2; if the ratio is very large, then the competitive ratio is close to 1.

5.2 Randomized controller

In a different methodology, we assume that θ_t is determined by a randomized algorithm that returns a value based on certain probability distribution P . During on-line control, the controller essentially is playing a game with an adversary (i.e., the task arrival process). Our job is to find the optimal probability distribution and the corresponding competitive ratio. We point out that the competitive ratio of a randomized on-line algorithm A is defined with respect to a specific type of adversary. In this paper, we assume an oblivious adversary (Ben-David et al. 1994), in which the worst instance for the randomized algorithm A is chosen without the the knowledge of the realization of the random variable used by A . We say randomized algorithm A is c -competitive if $E_P[\tilde{J}(A, I)] \leq cJ^*(I)$, $\forall I \in \mathcal{I}$, where $\tilde{J}(A, I)$ is the cost of algorithm A under task arrival instance I in on-line control and $J^*(I)$ is the corresponding off-line optimal cost. Note that the task arrival instance I must

be fixed before the expectation is taken. The competition ratio of randomized algorithm A_P (algorithm A using probability distribution P) is: $c(A_P) = \sup_{I \in \mathcal{I}} \frac{E_P[\tilde{J}(A_P, I)]}{J^*(I)}$. Our goal is to find the best possible probability distribution that yields the best competitive ratio $c^* = \inf_P \sup_{I \in \mathcal{I}} \frac{E_P[\tilde{J}(A_P, I)]}{J^*(I)}$. This is essentially a minimax problem, and one way of solving it is to use Yao’s minimax principle (Yao 1977), which states: a randomized algorithm may be viewed as a random choice between deterministic algorithms; in particular, the competitive ratio of a randomized algorithm against any oblivious adversary is the same as that of the best deterministic algorithm under the worst-case distribution of the adversary’s input. In our case, the adversary’s input is the task arrival instance after each AP. Let its probability distribution be G . Using Yao’s principle and von Neumann minimax theorem, we get: $c^* = \sup_G \inf_{A \in \mathcal{A}} \frac{E_G[\tilde{J}(A, I_G)]}{J^*(I_G)}$, where \mathcal{A} is the set of all randomized algorithms, I_G is a specific task arrival instance under probability distribution G , and the expectation is now performed with respect to G . We now use the following lemma to find c^* .

Lemma 2 *The best competitive ratio c^* of a randomized controller is obtained when θ_t is a random variable X , whose probability density function is*

$$f_X(x) = \begin{cases} \frac{1}{C_I(e-1)} e^{x/(C_W/C_I)}, & \text{if } x \leq C_W/C_I \\ 0, & \text{if } x > C_W/C_I \end{cases}$$

When this controller is used, $\lim_{N \rightarrow \infty} c^* = (\gamma + 1.58)/(\gamma + 1)$, where $\gamma = C_B\beta/C_W$.

Lemma 2 shows that the competition ratio of a random controller also depends on the ratio between $C_B\beta$ and C_W . If this ratio is very small, then the competitive ratio is close to 1.58; if the ratio is very large, then the competitive ratio is close to 1.

6 Conclusions

In this paper, we study the optimal ON-OFF control problem for a class of DESs with real-time constraints. The DESs have operating costs C_B and C_I per unit time and wake-up cost C_W . Our goal is to switch the system between the ON and the OFF states so as to minimize cost and satisfy real-time constraints. In particular, we consider a homogeneous case that all tasks have the same number of operations and each one’s deadline is d seconds after the arrival time. For the off-line scenario that all task information is known to the controller a priori, we show that the optimal solution can be obtained via a two-fold decomposition: (i) super active periods that contain one or more active periods can be identified easily using the task arrival times and deadlines and (ii) the optimal solution to each super active period can be solved using dynamic programming. Simulation results show that compared with a simple heuristic, the cost saving of the DP algorithm can be 50% or more.

In on-line control, we show that the best time to start an AP can be obtained via an iterative algorithm and is guaranteed to be the same as the off-line problem. To decide the best time to end an AP in the on-line setting where no future task arrival information is available, we evaluate both deterministic and random controllers and derive their competitive ratios; these results quantify the worst-case on-line performance deviation from the optimal off-line solution.

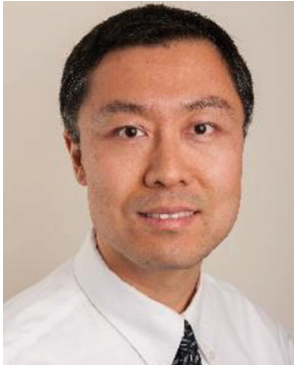
Acknowledgements We would like to thank Dallas Leitner for his help in generating the off-line simulation results. We also would like to thank the anonymous reviewers for the constructive feedback they provided in the review process.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

- Ben-David S, Borodin A, Karp R, Tardos G, Wigderson A (1994) On the power of randomization in on-line algorithms. *Algorithmica* 11(1):2–14
- Chen W, Mitra U, Neely M (2009) Energy-efficient scheduling with individual packet delay constraints over a fading channel. *ACM Wireless Networks* 15:601–618
- Chen W, Neely M, Mitra U (2007) Energy efficient scheduling with individual packet delay constraints: offline and online results. In: *IEEE Infocom*, Anchorage
- Cohen R, Kapchits B (2009) An optimal wake-up scheduling algorithm for minimizing energy consumption while limiting maximum delay in a mesh sensor network. *IEEE/ACM Trans Networking* 17:570–581
- Gamal AE, Nair C, Prabhakar B, Uysal-Biyikoglu E, Zahedi S (2002) Energy-efficient scheduling of packet transmissions over wireless networks. In: *Proceedings of IEEE INFOCOM*, vol 3, 23–27. New York City, pp 1773–1782
- Lai W, Paschalidis IC (2006) Routing through noise and sleeping nodes in sensor networks: latency vs. energy trade-offs. In: *The 45th IEEE conference on decision and control*, San Diego, pp 2716–2721
- Liu JWS (2000) *Real - Time Systems*. NJ, Prentice Hall Inc.
- Mao J, Cassandras CG (2014) Optimal control of multilayer discrete event systems with real-time constraint guarantees. *IEEE Trans Syst Man Cybern Syst Hum* 44(10):1425–1434
- Mao J, Cassandras CG, Zhao Q (2007) Optimal dynamic voltage scaling in energy-limited nonpreemptive systems with real-time constraints. *IEEE Trans Mob Comput* 6(6):678–688
- Miao L (2017) Optimal on-off scheduling for a class of discrete event systems with real-time constraints. In: *American control conference*, Seattle
- Miao L, Mao J, Cassandras CG (2017) Optimal Energy-Efficient Downlink Transmission Scheduling for Real-Time Wireless Networks. *IEEE Trans Control Netw Syst* 4(4):692–706
- Miao L, Xu L (2015) Optimal wake-up scheduling for energy efficient fixed-rate wireless transmissions with real-time constraints. In: *14th wireless telecommunications symposium*, New York
- Miao L, Xu L, Jiang D (2018) Optimal On-Off Control for a Class of Discrete Event Systems with Real-Time Constraints. [arXiv:1809.01711](https://arxiv.org/abs/1809.01711)
- Ning X, Cassandras CG (2006) Dynamic sleep time control in event-driven wireless sensor networks. In: *The 45th IEEE conference on decision and control*, San Diego, pp 2722–2727
- Ning X, Cassandras CG (2008) Optimal dynamic sleep time control in wireless sensor networks. In: *The 47th IEEE conference on decision and control*, Cancun, pp 2332–2337
- Pepyne DL, Cassandras CG (2000) Optimal control of hybrid systems in manufacturing. *Proc IEEE* 88(7):1108–1123
- Poulakis MI, Panagopoulos AD, Constantinou P (2013) Channel-aware opportunistic transmission scheduling for energy-efficient wireless links. *IEEE Trans Veh Technol* 62:192–204
- Uysal-Biyikoglu E, Prabhakar B, Gamal AE (2002) Energy-efficient packet transmission over a wireless link. *IEEE/ACM Trans Networking* 10:487–499
- Wang X, Li Z (2013) Energy-efficient transmissions of bursty data packets with strict deadlines over time-varying wireless channels. *IEEE Trans Wirel Commun* 12:2533–2543
- Xia L, Miller D, Zhou Z, Bambos N (2017) Service rate control of tandem queues with power constraints. *IEEE Trans Autom Control* 62(10):5111–5123
- Yao AC-C (1977) Probabilistic computations: toward a unified measure of complexity. In: *18th annual symposium on foundations of computer science*, 1977, IEEE, pp 222–227
- Ye W, Heidemann JS, Estrin D (2004) Medium access control with coordinated adaptive sleeping for wireless sensor networks. *IEEE/ACM Trans Networking* 12:493–506
- Zafer M, Modiano E (2007) Delay-constrained energy efficient data transmission over a wireless fading channel. In: *IEEE Information theory and applications workshop*, San Diego
- Zafer M, Modiano E (2008) Optimal rate control for delay-constrained data transmission over a wireless channel. *IEEE Trans Inf Theory* 54:4020–4039

- Zafer M, Modiano E (2009) A calculus approach to energy-efficient data transmission with quality-of-service constraints. *IEEE/ACM Trans Networking* 17:898–911
- Zafer M, Modiano E (2009) Minimum energy transmission over a wireless channel with deadline and power constraints. *IEEE Trans Autom Control* 54:2841–2852
- Zhong X, Xu C (2008) Online energy efficient packet scheduling with delay constraints in wireless networks. In: *IEEE Infocom*, Phoenix



Lei Miao received the B.S. and M.S. degrees from Northeastern University, Shenyang, Liaoning, China, and the Ph.D. degree from Boston University, Boston, MA, in 1998, 2001, and 2006, respectively. From 2006 to 2009, he was with Nortel Networks in Billerica, MA. From 2009 to 2011, he was a visiting professor at the University of Cincinnati. From 2011 to 2014, he was with NuVo Technologies/Legrand North America in Hebron, KY. From 2014 to 2015, he was a faculty member at State University of New York Farmingdale. Currently, he is Assistant Professor of Mechatronics Engineering at Middle Tennessee State University. His research interests include control and optimization for Discrete Event Systems and hybrid systems, with applications to communication networks, wireless networks, and cyber-physical systems.



Lijian Xu (S'11M'14) received the B.S. degree from University of Science and Technology, Beijing, China, in 1998; the M.S. degree from University of Central Florida, Orlando, FL, USA, in 2001; and the Ph.D. degree from Wayne State University, Detroit, MI, USA, in 2014. From 2001 to 2007 he was an Engineer and an Engineering Manager with AT&T, Navarre, FL, and Telus Communication, Calgary, AB, Canada, Communication, Canada, respectively. He is currently an Assistant Professor in the Department of Electrical and Computer Engineering Technology, Farmingdale State College, State University of New York, Farmingdale, NY, USA. His interests are in the areas of network control system, digital wireless communication, consensus control, and vehicle platoon and safety. Dr. Xu received the Best Paper Award from the 2012 IEEE International Conference on Electro/Information Technology.



Dingde Jiang (S'08-M'09) received the Ph.D. degree in communication and information systems from the University of Electronic Science and Technology of China, Chengdu, China, in 2009.

From 2013 to 2014, he was a Visiting Scholar with the Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN, USA. He was a Professor with the School of Computer Science and Engineering and the College of Information Science and Engineering at Northeastern University, Shenyang, China, before April of 2017. He is currently a Professor with the School of Astronautics and Aeronautic, University of Electronic Science and Technology of China. His research focuses on network measurement, modeling and optimization, performance analysis, network management, network security in communication networks, particularly in software-defined networks, information-centric networking, energy-efficient networks, and cognitive networks. His research is supported by the National Science Foundation of China, the Program for New Century Excellent Talents with the University of Ministry of Education of China, and so on.

Dr. Jiang served as a technical program committee member for several international conferences. He has received the best paper awards at several international conferences. He has been serving as an editor for one international journal.