Optimal Dynamic Buffer Management
Using Optimal Control of Hybrid Systems

Wei Zhang, Jianghai Hu *

Department of Electrical and Computer Engineering, Purdue University, West Lafayette, IN47906, USA

Abstract
This paper studies a general dynamic buffer management problem with one buffer inserted between two interacting components. The component to be controlled is assumed to have multiple power modes corresponding to different data processing rates. The overall system is modeled as a hybrid system and the buffer management problem is formulated as an optimal control problem. The objective function of the proposed problem depends on the switching cost and the size of the continuous state space, making its solutions much more challenging. By exploiting some particular features of the proposed problem, the best mode sequence and the optimal switching instants are characterized analytically using some variational approach. Simulation result based on real data shows that the proposed method can save about 60% of energies compared with another heuristic scheme in several typical situations.

Key words: Hybrid systems, Optimal control, Dynamic buffer management, Variational method, Low power design of embedded systems

1 Introduction

Dynamic buffer management (DBM) is an effective power management technique that can reduce the power consumptions of electronic devices by inserting buffers among interacting components. The buffer insertion makes it possible to turn off underutilized component at appropriate times without affecting the service for the other components, thus reducing the system power consumption. The optimal buffer size resulting in the largest power reduction is derived in Cai and Lu [2005], Hu and Lu [2005], Ridenour et al. [2007] for some simple DBM problems. A major limitation of these studies is that they all assume that the components to be controlled have only two power modes, “on” and “off”. However, in practice, many components can work in more than two power modes, such as the variable speed processors (Burd and Brodersen [2000]) and the multi-speed disks (Gurumurthi et al. [2003]). For such a component, instead of completely turning it off, we can properly design a switching strategy, namely the scheduling of different power modes of the component, to further reduce its power consumption.

Optimal control of hybrid systems is a challenging research topic that has attracted many researchers. In Branicky et al. [1998], a unified framework is formulated for optimal control of hybrid systems; some conceptual algorithms based on the Bellman equation are also proposed for computing the optimal control policies. A similar idea is employed in Hedlund and Rantzer [1999], where a more detailed algorithm based on the discretization of the continuous state space is developed to solve the Bellman inequality. In Xu and Antsaklis [2000a,b], a two-stage optimization method is proposed for switched systems, where in the first stage the optimal continuous input is computed for a fixed switching strategy and then in the second stage the dynamic programming algorithm is used to compute the best switching strategy. In parallel with these dynamic-programming-based approaches, variational methods have also been extensively studied. In Riedinger et al.
[1999a,b], the maximum principle is generalized to solve a time optimal control and a linear quadratic control problem for switched systems with linear subsystems. Some more general versions of the maximum principle for hybrid systems are proved in Sussmann [1999] and Piccoli [1998]. Variational approaches are also used in Xu and Antsaklis [2004], Egerstedt et al. [2006] to derive necessary conditions for the optimal switching instants and/or the optimal continuous control input for switched systems with a fixed switching sequence. Although an algorithm for updating the switching sequence is discussed in Egerstedt et al. [2006], finding the best switching sequence is still an NP-hard problem. More recently, Bengea and DeCarlo [2005] propose a way of embedding a switched system into a larger family of systems, whose solutions, obtained by the traditional optimal control methods, can be used to construct the optimal control of the switched systems without enumerating the switching sequences. Besides these theoretical works, applications of the optimal control theory of hybrid systems in various practical contexts have also been well studied. The problems in this category as in Cassandras et al. [2001], Wanga et al. [1997], Gokbayrak and Cassandras [2000a], Gokbayrak and Cassandras [2000b], usually deal with particular model structures and cost functions that often enable one to find better analytical and numerical solutions. The optimal control problem considered in this paper falls into this category.

Despite the richness of the literature in this field, the problem studied in this paper can not be directly solved using the existing methods as it has the following distinct features: (i) transitions among discrete modes depend on the evolution of the continuous state; whereas many previous studies ignore such dependence; (ii) the switching (mode) sequence is a decision variable that cannot be assumed fixed as in Xu and Antsaklis [2004] and Egerstedt et al. [2006]; (iii) the switching cost ignored in most previous papers is an important part of our cost function; (iv) the buffer size that determines the range of the continuous states is variable, indicating that both the optimal control and the optimal size of the continuous state space are to be designed at the same time. Few existing results have addressed all of these issues.

The main contributions of this paper are the following: (i) Hybrid system framework is successfully applied to model the DBM problem, which is an important problem in the low power design of embedded systems. (ii) Two practically important DBM problems are formulated as optimal control problems of a piecewise-constant hybrid system and solved analytically through a variational approach. (iii) Several issues of implementing the proposed optimal strategy in practical systems are addressed. The results are also verified through some simulations based on real data.

The rest of this paper is organized as follows. In Section 2, two DBM problems are introduced and formulated as optimal control problems of a piecewise constant hybrid system. In Section 3, several operations on hybrid trajectories are introduced. These operations are then used in Sections 4 and 5 to derive the optimal solutions. Two simulation examples are given in Section 6 to illustrate the effectiveness of the optimal strategies. Concluding remarks and future research directions are discussed in Section 7.

2 Problem Formulation

2.1 System Description

Consider two interacting components X and Y as shown in Fig. 1, where X produces data for Y to consume. Suppose that Y is always “on” and consumes data at a constant speed $r_y$. On the other hand, assume that X has N different operation modes where in mode $i$, $i = 1, 2, \ldots, N$, it produces data at a constant speed $r_i$ and consumes power $p_i$. Without loss of generality, assume $r_1 < r_2 < \cdots < r_N$. Usually, a lower data rate corresponds to a lower power consumption; thus we require $p_1 < p_2 < \cdots < p_N$. Denote by $I$ and $J$ the sets of indices whose corresponding data rates are greater and smaller than $r_y$, respectively, i.e.,

$$I = \{i \mid r_i > r_y, i = 1, \ldots, N\},$$

and

$$J = \{j \mid r_j < r_y, j = 1, \ldots, N\}.$$

Assume that both $I$ and $J$ are nonempty, i.e., $r_N > r_y > r_1$. Note that we ignore the degenerate case where $r_y$ can be perfectly matched by one of the power modes of X, since in this case no buffer is needed and the DBM problem becomes meaningless. A mode $\sigma$ is called an ascending mode if $\sigma \in I$ and a descending mode otherwise. To ensure smooth operation, a buffer B with capacity $Q$ is inserted between X and Y. See Fig. 1 for the configuration of the overall system.

Many real-world applications can be described by the above system. One simple example is the data-copying process, where a device Y copies data from a hard drive X. The hard drive has two power modes “on” and “off”. If the data rate of the hard drive is faster than that of Y, then X can be turned off during some time intervals to save energy. In this case, the system memory,
which serves as the buffer B in our model, is needed to temporarily store the data from X for later delivery. As another example, consider the video playing process. Let X be the Intel Xscale processor (Lu et al. [2002]) that can operate on multiple voltages corresponding to different speeds $r_i$’s and powers $p_i$’s; let Y be a video card that demands data from X at a constant speed, say 30 frame/sec. To ensure smooth operation, the system memory is needed as a buffer to store the data that has been decoded by X but yet to be displayed by Y. Thus, the abstract system as shown in Fig. 1 represents a class of practical systems. Minimizing the power consumption of such a system has important practical implications.

2.2 Hybrid System Model

The above problem can be modeled as a hybrid system $H$. The discrete state space of $H$ consists of $N$ modes: $S = \{1, 2, \ldots, N\}$, corresponding to the operation modes of X. The continuous state $q(t)$ is defined as the amount of data stored in the buffer B, and is thus required to take values in the interval $[0, Q]$. The evolution of $q(t)$ is determined by the speed difference between the two components, i.e., $q(t) = r_i - r_j$ for mode $i$. As a physical constraint, there can be no buffer underflow or overflow. Thus, we require that whenever $q(t)$ hits the boundary of its domain, namely, $q(t) = 0$ or $Q$, the system must transit to another mode that can bring $q(t)$ back to the inside of $[0, Q]$. Except for this, there are no other transition rules and guard conditions. The reset map of the system is trivial, i.e., there is no jump in $q(t)$ at the transition instant.

Given a time period $[0, t_f]$, the behavior of the above system can be uniquely determined by the switching strategy $\sigma : [0, t_f] \to S$, which determines the active mode of the system over $[0, t_f]$. The overall trajectory $z(t) = (q(t), \sigma(t))$ of the hybrid system consists of the trajectories of the continuous state $q(t)$ and the discrete state $\sigma(t)$. For a given initial value $q(0)$, the system is governed by the following differential equation:

$$\frac{dq(t)}{dt} = r_{\sigma(t)} - r_y, \quad \forall t \in [0, t_f].$$ (1)

In this paper, we study the power consumption of the whole process of transferring a certain amount of data from X to Y. It is thus required that the system must start with an empty buffer at $t = 0$ and end up with an empty buffer at $t = t_f$ when Y have received all the data produced by X. This yields two boundary conditions for the continuous state, namely, $q(0) = 0$ and $q(t_f) = 0$. The hybrid trajectories that satisfy these two conditions are called feasible trajectories (See Fig. 2-(a)).

We assume that there is a partition of $[0, t_f]$, $t_0 = 0 \leq t_1 \leq \ldots \leq t_n = t_f$, for some $n \geq 0$, so that $\sigma(t) \equiv \sigma_i \in S$ is constant in each subinterval $[t_{i-1}, t_i)$, $i = 1, \ldots, n$. The sequence $(\sigma_1, \ldots, \sigma_n)$ is called the switching sequence and $(t_0, \ldots, t_{n-1})$ is called the switching instants$^1$.

A hybrid trajectory $z(t) = (q(t), \sigma(t))$ over $[0, \infty)$ is called periodic with period $T$ if $q(t + T) = q(t)$ and $\sigma(t + T) = \sigma(t)$ for all $t \in [0, \infty)$. For such a trajectory, denote by $n_T$ the number of switchings in each period. For example, $n_T = 5$ for the trajectory in Fig 2-(c).

A feasible trajectory is called a $\Lambda$-trajectory if it consists of one ascending mode $i$ and one descending mode $j$ with exactly two switchings as shown in Fig 2-(d). The pair of modes $\{i, j\}$ in a $\Lambda$-trajectory is called a $\Lambda$-pair.

A feasible trajectory $z(t) = (q(t), \sigma(t))$ with switching instants $(t_0, \ldots, t_{n-1})$ is called a boundary-switching trajectory (BST) if $q(t_i) = Q$ or 0 for any $i = 0, \ldots, n-1$. In other words, a BST only switches at the boundary of the range of $q(t)$. Denote by $\Omega$ the class of all BSTs. Every BST can be decomposed into a series of $\Lambda$-trajectories with the same buffer size. Denote by $n_\Lambda$ the number of distinct $\Lambda$-pairs in a BST. For example, $n_\Lambda = 3$ for the BST in Fig 2-(e). A BST is called pure if $n_\Lambda = 1$ and is called mixed otherwise. In other words, a pure trajectory must be a BST and is obtained by repeating a $\Lambda$-trajectory for a certain number of times (See Fig. 2-(f)).

The power consumption of a given hybrid trajectory

$^1$ The system is turned on at $t = 0$. Hence, we assume that there is always a switching at $t = 0$ and ignore the switching, if any, at $t = t_f$ for all trajectories.
z(t) = (q(t), \sigma(t)) consists of three parts: the running power, namely the power consumed by component X, the switching power and the buffer power. Note that \( p_{\sigma(t)} \) is the instantaneous power of X at time t. Thus the average running power over \([0, t_f]\) is \( \frac{1}{t_f} \int_0^{t_f} p_{\sigma(t)} dt \). Assume that switching among different modes consumes the same amount of energy \( k_s \). Then the average switching power over \([0, t_f]\) is \( nk_s/t_f \), where \( n \) is the number of switchings in the trajectory \( z \). The buffer power includes the static buffer power and the dynamic buffer power. The static buffer power is proportional to the buffer size while the dynamic buffer power only depends on the actual amount of data in the buffer. Since the dynamic buffer power is much smaller than the static one, in this paper, we only consider the static buffer power and denote it by \( p_b Q \), where \( p_b \) is a positive constant and \( Q \) is the buffer size. Thus the total average power of the system during \([0, t_f]\) can be written as

\[
P(z; Q, t_f) = \frac{1}{t_f} \int_0^{t_f} p_{\sigma(t)} dt + \frac{nk_s}{t_f} + p_b Q,
\]

and the total energy associated with \( z(t) \) during \([0, t_f]\) is

\[
E_\sigma(z; Q, t_f) = \int_0^{t_f} p_{\sigma(t)} dt + nk_s + p_b Q \cdot t_f.
\]

The three terms on the right hand side of the above equation represent the running energy, the switching energy, and the buffer energy, respectively.

### 2.3 Problem Statements

The goal of this paper is to find a feasible trajectory that can finish a given task with the least energy consumption. For some applications, we know exactly the amount of data to be transferred to Y. In this case, \( t_f \) is a given constant which equal to the amount of data to be transferred divided by the data rate of Y. The energy minimization problem can be formulated as the following optimal control problem of hybrid system H.

**Problem 1** \( \min_{z,Q} P(z;Q,t_f) \) subject to the constraints: (i) \( z(t) = (q(t), \sigma(t)) \) satisfies equation (1); (ii) \( q(t) \in [0, Q], \forall t \in [0, t_f] \) and \( q(0) = q(t_f) = 0 \); (iii) \( \sigma(t) \in S, \forall t \in [0, t_f] \).

Problem 1 requires the exact knowledge of \( t_f \). However, in some applications, the time horizon \( t_f \) is not known a priori. For example, consider that a network card (component X) downloads a movie from the internet and at the same time sends the data to a video card (component Y) to play. The \( t_f \) in this example is determined by the length of the movie and may not be known until X receives the last frame. In this case, we are usually interested in periodic strategies that are easy to implement and whose power can be computed even without the knowledge of \( t_f \). Therefore, another meaningful problem is to find the optimal periodic trajectory with the least average power consumption.

Let \( z(t) \) be a periodic trajectory with \( (\sigma_1, \ldots, \sigma_{n_T}) \) as the switching sequence and switching instants during the first period \([0, T]\), respectively. Note that the periodic trajectory has an infinite length, i.e., \( t_f = \infty \). The average power of \( z \) is the same as its average power during the first period, i.e.,

\[
P(z; Q, \infty) = P(z; Q, T) = \frac{1}{T} \left( \sum_{i=1}^{n_T} p_{\sigma_i} \tau_i + nT k_s \right) + p_b Q.
\]

Since every feasible solution must start with zero buffer, we must have \( q(T) = q(0) = 0 \), i.e., at the end of each period, continuous state \( q \) must come back to zero. Different from Problem 1, to find the best periodic solution, we not only need to optimize the switching sequence and switching instants, but also need to find the best period \( T \). This is formulated as the following problem.

**Problem 2** \( \min_{z,Q,T} P(z;Q,T) \) subject to the constraints: (i) \( z(t) = (q(t), \sigma(t)) \) is periodic with period \( T \) and satisfies equation (1); (ii) \( q(t) \in [0, Q], \forall t \in [0, T] \) and \( q(0) = q(T) = 0 \); (iii) \( \sigma(t) \in S, \forall t \in [0, T] \).

**Remark 3** The two problems in this section are independent to each other and serve different purposes. Problem 1 is suitable for the case where the value of \( t_f \) is known exactly before the system starts operating. On the other hand, for unknown \( t_f \), Problem 2 prepares for the worst case by assuming \( t_f = \infty \) and only focuses on infinite-length periodic strategies. However, for real applications the time horizon \( t_f \) must be finite. Thus, when the solution of Problem 2 is applied to a real system, only part of the strategy will be used. See Section 4.3 for implementation details of periodic strategies.

For any optimal solutions to Problem 1 and 2, to avoid the unnecessary power consumption by the unused buffer space, \( Q \) should be chosen as small as possible so that the buffer is full at least once during \([0, t_f]\). In addition, since \( q(t) \geq 0 \) and \( q(0) = 0 \), the following lemma follows immediately.

**Lemma 1** If \( z(t) = (q(t), \sigma(t)) \) is an optimal solution to Problem 1, we must have

\[
\min_{t \in [0, t_f]} q(t) = 0, \quad \text{and} \quad \max_{t \in [0, t_f]} q(t) = Q.
\]
This condition also holds for Problem 2 with $t_f$ replaced by $T$.

According to Lemma 1, the optimal buffer size is completely determined by a given trajectory $z(t)$. From now on, we will call $Q$ a valid buffer size of $z$ if $Q \leq \max q(t)$ and the optimal buffer size of $z$ if the equality holds.

The rest of this paper is devoted to deriving analytical solutions to the two problems formulated in this section. Specifically, we will prove that: (i) the optimal solutions to both problems must be boundary-switching trajectories (BSTs); (ii) the optimal pure periodic trajectory (OPPT) with $n_p = 1$ is an optimal solution to Problem 2 for an arbitrary $n_p$; (iii) the optimal pure trajectory (OPT) with length $t_f$ and $n_p = 1$ is an optimal solution to Problem 1 for an arbitrary $n_p$; (iv) the OPT is different from the OPPT in general and will converge to the OPPT as $t_f$ goes to infinity. Although we consider all feasible trajectories as candidate solutions, the above results enable us to only focus on pure (periodic) trajectories in finding the optimal solutions. Since a pure trajectory involves only one (distinct) $\Lambda$-pair and only switches when $q(t)$ is 0 or $Q$, the OPT and OPPT, which are optimal solutions to Problems 1 and 2, can be easily characterized analytically.

3 Operations on Hybrid Trajectories

In this section, we introduce some important operations that can transform an existing trajectory to a new one while preserving certain properties. These operations play an important role in deriving the optimal solutions to Problems 1 and 2.

3.1 Cropping

Cropping, denoted by $C_{a,b}[\cdot]$, is an operation that obtains a new trajectory by trimming off the uninteresting parts of the original trajectory. For example, the cropped trajectory $C_{a,b}[z]$ will only keep the part of $z(t)$ where $t \in [a,b]$, i.e.,

$$C_{a,b}[z](t) = z(t + a), \quad \text{for} \ t \in [0, b - a].$$

3.2 Joining

Joining, denoted by $J[\cdot; \cdots; \cdot]$, is an operation that obtains a new trajectory by putting several finite-length trajectories together. For example, $J[z^{(1)}, z^{(2)}]$ corresponds to a new trajectory obtained by appending $z^{(2)}$ to the end of $z^{(1)}$. More precisely,

$$J[z^{(1)}, z^{(2)}](t) = \begin{cases} 
z^{(1)}(t) & t \leq t_f^{(1)} \\
z^{(2)}(t - t_f^{(1)}) & t > t_f^{(1)}
\end{cases},$$

where $t_f^{(1)}$ and $t_f^{(2)}$ are the lengths of $z^{(1)}$ and $z^{(2)}$, respectively. To prevent introducing discontinuities, it is required that $z^{(1)}(t_f^{(1)})$ and $z^{(2)}(t_f^{(1)})$ have consistent boundary conditions, i.e., $q^{(1)}(t_f^{(1)}) = q^{(2)}(0)$, where $q^{(1)}$ and $q^{(2)}$ are the continuous states of $z^{(1)}$ and $z^{(2)}$, respectively. Denote by $J_m[z]$ a special joining operation that repeats the trajectory $z$ for $m$ times, i.e.,

$$J_m[z] = J[z, \ldots, z].$$

Again, the argument $z$ in the above operation must satisfy $z(0) = z(t_f)$.

3.3 Periodic Extension

Periodic extension, denoted by $P[\cdot]$, is an operation that obtains a periodic trajectory by repeating a given trajectory $z$ for infinitely many times. Mathematically, $P[\cdot]$ can be defined in terms of the joining operation as $P[z] = J_{\infty}[z]$. For a trajectory $z(t)$ of length $t_f$, $P[z](t + t - t_f)$ is periodic for all $t \in [0, t_f]$ and any nonnegative integer $l$.

3.4 Scaling

For an arbitrary hybrid trajectory $z(t) = (q(t), \sigma(t))$, the scaling operation with parameter $c > 0$ is defined as

$$S_c[z](t) = (cq(t/c), \sigma(t/c)).$$

If $z(t) = (q(t), \sigma(t))$ is a hybrid trajectory in $[0, t_f]$ with buffer size $Q$, switching sequence $(\sigma_1, \ldots, \sigma_n)$ and switching instants $(t_0, \ldots, t_{n-1})$, then $S_c[z]$ is a hybrid trajectory in $[0, ct_f]$ with buffer size $cQ$, switching sequence $(\sigma_1, \ldots, \sigma_n)$ and switching instants $(ct_0, \ldots, ct_{n-1})$. In other words, $S_c[z]$ follows exactly the same switching sequence as $z$, but the time it spends in each mode before switching to a new one is scaled by a factor of $c$. An important property of the scaling operation is that it does not change the running power of a trajectory. It can be easily verified that the total average power of $S_c[z]$ is

$$P(S_c[z]; cQ, ct_f) = P(z; Q, t_f) + \frac{(1 - c)nk_s}{ct_f} + p_b(c - 1)Q.$$

3.5 Folding

Folding operation, denoted by $F_m[\cdot]$, is only defined for the $\Lambda$-trajectories. It obtains a pure trajectory with $2m$ switchings from a $\Lambda$-trajectory with $2$ switchings. Fig. 3 illustrates a 3-fold folding for a $\Lambda$-trajectory $z(t)$. The operation consists of the following three steps. First, the
3.6 Switching Instant Perturbation (SIP)

Switching instant perturbation is defined only for two-switching trajectories, namely the trajectories (may not be feasible in general) with exactly two switchings. Let $z(t) = (q(t), \sigma(t))$ be a two-switching trajectory with switching sequence $(\sigma_1, \sigma_2)$, switching instants $(t_0, t_1)$ and buffer size $Q$. Suppose that $q(0) = q_1$ and $q(t_f) = q_2$. Denote by $\mathcal{H}(z) = (\hat{q}(t), \hat{\sigma}(t))$ the SIP on $z$. Roughly speaking, the SIP is an operation that perturbs the switching instant $t_1$ to a neighboring value $\hat{t}_1$ while at the same time changes the time $t_f$ accordingly to a certain value $\hat{t}_f$ to maintain the same trajectory boundary values, i.e., $\hat{q}(0) = q_1$ and $\hat{q}(\hat{t}_f) = q_2$. Fig. 4 illustrates an example of obtaining $\mathcal{H}(z)$ from $z(t)$. It is observed that the new trajectory $\mathcal{H}(z)$ switches from mode $\sigma_1$ to mode $\sigma_2$ at time $h$ instead of $t_1$ and ends at time $\hat{t}_f$ when its continuous state hits $q_2$. Mathematically, the range of $z(t)$ is divided evenly into 3 sections. Each section corresponds to two segments of the trajectory: one is ascending and the other one is descending. Then by appending each descending segment to the end of the corresponding ascending one, a new $\Lambda$-trajectory is obtained in each section. Finally, all the three $\Lambda$-trajectories are joined together to obtain the final trajectory as shown in Fig. 3-(c).

If $Q$ is the optimal buffer size of $z(t)$, then $\mathcal{F}_m[z]$ contains $2m$ switchings and has an optimal buffer size $Q/m$. In fact, any finite-length pure trajectory (e.g. Fig. 3)-(c)) through the folding operation with certain parameter $m$.

Fig. 4. Switching instant perturbation $\mathcal{H}(z)$ with $h < t_1$.

SIP can be defined as

$$\mathcal{H}_h[z] = (\hat{q}(t), \hat{\sigma}(t)),$$

where

$$\hat{\sigma}(t) = \begin{cases} \sigma_1 & t \leq h \\ \sigma_2 & h < t \leq \hat{t}_f \end{cases},$$

$$\frac{dq(t)}{dt} = r_\sigma(t) - r_y, \quad \text{for } t \in [0, \hat{t}_f],$$

and

$$\hat{t}_f = h + \frac{h(r_y - r_{\sigma_2}) + q_2 - q_1}{r_{\sigma_2} - r_y}.$$  

Under the above notations, a SIP $\mathcal{H}_h[z]$ is called valid if

$$0 \leq h \leq \hat{t}_f,$$

and

$$\hat{q}(t) \in [0, Q] \quad \text{for all } t \in [0, \hat{t}_f].$$  

In other words, $\mathcal{H}_h[z]$ is valid if it spends nonnegative time in each mode and it does not cause any buffer overflow or underflow. The set of $h$ for which $\mathcal{H}_h[z]$ is valid is called the domain of $h$ and is denoted by $D_h$. Thus for any $h \in D_h$, $\hat{z} = \mathcal{H}_h[z]$ defined in (4) satisfies the following properties:

1. $\hat{z}$ follows the same switching sequence $(\sigma_1, \sigma_2)$ as $z$ and spends nonnegative time in each mode.
2. $\hat{q}(0) = q(0) = q_1$ and $\hat{q}(\hat{t}_f) = q(t_f) = q_2$.
3. $\hat{q}(t) \in [0, Q]$ for all $t \in [0, \hat{t}_f]$.

Note that $D_h$ is a bounded connected interval. For example, consider the trajectory $z_1$ as shown in Fig. 5-(a). Let $(\hat{q}(t), \hat{\sigma}(t)) = \mathcal{H}_h[z_1](t)$. If $h < a$, then $\hat{t}_f$ as defined in (4) will be less than $h$, which violates the first condition in (5). On the other hand, if $h > b$, then $\hat{t}_f < 0$ for $t \in (b, \hat{t}_f)$, which violates the second condition in (5). Hence, $D_h = [a, b]$ for $z_1$. As another example, consider the trajectory $z_2$ as shown in Fig. 5-(b) for which $q(t_1)$ is on the boundary of $[0, Q]$. By a similar argument as in the first example, the range of $h$ for $z_2$ is $D_h = [c, t_1]$. It is observed from these two examples that if $q(t_i) \in (0, Q)$, then $t_i$ is an interior point of $D_h$. On the other hand, if $q(t_i) = 0$ or $Q$, then $t_i$ is on the boundary of $D_h$. This
property actually holds for the SIP of arbitrary two-switching trajectories.

The SIP is a specific yet useful operation. Since it can perturb the switching instant without affecting the boundary values \((q(0)\) and \(q(T)\)) and the buffer size \(Q\), it can be used, together with other operations such as cropping and joining, to study the effect of perturbing only one switching instant of a general trajectory.

### 4 Optimal Periodic Solution

In this section, we derive optimal solutions of Problem 2 (OS2). The following lemma can greatly simplify the problem and is crucial for later proofs.

**Lemma 2** If \(z\) is an OS2, then \(z \in \Omega\). In other words, optimal solutions to Problem 2 must be boundary-switching trajectories.

**PROOF.** The key idea of the proof is to use the operations defined in Section 3 to construct a better trajectory with less power consumption for any given trajectory that has switchings at some interior points of \([0, Q]\). Let \((z(t), Q, T)\) be a solution to Problem 2. Denote by \((\sigma_1, \ldots, \sigma_{n_T})\) and \((t_1, \ldots, t_{n_T})\) the switching sequence and switching instants in the first period of \(z(t)\). Suppose that \(z(t)\) has a switching at some interior point of \([0, Q]\), i.e., \(0 < q(t_i) < Q\) for some \(i\). Divide the first period of \(z(t)\) into three parts through the cropping operation as shown in Fig. 6-(a) and define them as three new trajectories.

\[
\begin{align*}
    z^{(1)}(t) &= C_{0, t_{i-1}}[z](t), \quad z^{(2)}(t) = C_{t_{i-1}, t_i}[z](t), \\
    z^{(3)}(t) &= C_{t_i, T}[z](t).
\end{align*}
\]

(6)

Assume that \(z^{(2)}(t) = (q^{(2)}(t), \sigma^{(2)}(t)), q^{(2)}(0) = q_1\) and \(q^{(2)}(t_i - t_{i-1}) = q_2\). Perform the SIP on \(z^{(2)}\) to obtain a new trajectory \(z_h^{(2)} = (q_h^{(2)}(t), \sigma_h^{(2)}(t)) \triangleq H_h[z^{(2)}]\). According to (4), the length of \(z_h^{(2)}\) is

\[
t_h^{(2)} = h + \frac{h(r_y - r_{\sigma_i}) + q_2 - q_1}{r_{\sigma_i+1} - r_y}. 
\]

(7)

By definition, the SIP does not change the boundary values of \(z^{(2)}\); i.e., \(q_h^{(2)}(0) = q_1\) and \(q_h^{(2)}(T) = q_2\). Thus we can rejoin \(z^{(1)}\), \(z_h^{(2)}\) and \(z^{(3)}\) as shown in Fig. 6-(b) to obtain

\[
z_h \triangleq \mathcal{J}[z^{(1)}, z_h^{(2)}, z^{(3)}].
\]

(8)

It is obvious that the length of \(z_h\) is

\[
T_h = t_{i-1} + t_h^{(2)} + (T - t_{i+1}).
\]

(9)

Now we show that \(z_h\) consumes less power than \(z\) for some \(h\). Recall that \(D_h\) is the set of \(h\) that \(z_h^{(2)}\) remains valid. According to (5), \(Q\) is a valid buffer size for \(z_h\) if \(h \in D_h\). Thus \(\forall h \in D_h\) the power of \(z_h\) with buffer size \(Q\) is

\[
\begin{align*}
    P(z_h; Q, T_h) &= \frac{1}{T_h} \left[ E_1 + E_3 + n_T k_s + p_{\sigma_i} \cdot h \\
    &+ p_{\sigma_{i+1}}(t_h^{(2)} - h) \right] + p_b Q.
\end{align*}
\]

(10)
Thus either $\frac{\partial P(z_h; Q, T_h)}{\partial h}$ has been cancelled out. Suppose that $D$ consumes less power than $z$ in $\Lambda$-pair $z$ than $z$. Theorem 2 enable one to focus on the BSTs ($\Omega$) in derivations of $\Lambda$-pairs. Recall that the Lemma 2 enables one to derive the optimal solutions to Problem 2. Let $\{i, j\}$ be a periodic trajectory with average power $Q$, $\sigma$ and $\delta$ as the switching sequence and switching instants in its first period, respectively. If $\{i, j\} \in \Sigma$ and $Q = Q^*_{ij}$ as defined in (15), then $z$ is an OPPT with period $T_{ij}$.

4.1 Optimal Pure Periodic Trajectory (OPPT)

The Optimal pure periodic trajectory (OPPT) is defined as the optimal periodic solution to Problem 2 under an additional constraint $n_p = 1$, i.e., the candidate trajectories must be pure boundary-switching trajectories. Let $z$ be a periodic trajectory satisfying this condition. Then every period of $z$ consists of the same $\Lambda$-pair. Thus the main task of this subsection is to find the best $\Lambda$-pair and the best period $T$ of $z$.

For a given $\Lambda$-pair $\{i, j\}$, the period $T_{ij}$ can be expressed in terms of the corresponding buffer size $Q_{ij}$ as

$$T_{ij} = \frac{Q_{ij}}{r_i - r_y} + \frac{Q_{ij}}{r_y - r_j} = \alpha_{ij} Q_{ij}. \tag{12}$$

Denote by $\beta_{ij}$ the running power of $z$ over a period, i.e.,

$$\beta_{ij} = \frac{1}{T_{ij}} \left[ \frac{Q_{ij} p_i}{r_i - r_y} + \frac{Q_{ij} p_j}{r_y - r_j} \right] = \frac{1}{\alpha_{ij}} \left[ \frac{p_i}{r_i - r_y} + \frac{p_j}{r_y - r_j} \right]. \tag{13}$$

Note that both $\alpha_{ij}$ and $\beta_{ij}$ are constants depending only on the given $\Lambda$-pair. With these notations, the average power of $z$ over one period is given by

$$\bar{P}_{ij}^\Lambda(Q_{ij}) = \beta_{ij} + \frac{2k_e}{\alpha_{ij} Q_{ij}} + p_h Q_{ij} \tag{14}$$

Taking the derivative of (14) with respect to $Q_{ij}$ and setting it to zero, we obtain the optimal buffer size for the $\Lambda$-pair $\{i, j\}$ as:

$$Q_{ij}^* = \sqrt{\frac{2k_e}{\alpha_{ij} p_h}}. \tag{15}$$

Thus the minimum achievable power for the $\Lambda$-pair $\{i, j\}$ is $\bar{P}_{ij}^\Lambda(Q_{ij}^*)$. The optimal $\Lambda$-pair $\{\sigma^+_{ij}, \sigma^-_{ij}\}$ can be obtained by minimizing $\bar{P}_{ij}^\Lambda(Q_{ij}^*)$ with respect to $\{i, j\}$, i.e.,

$$\{\sigma^+_{ij}, \sigma^-_{ij}\} = \arg \min_{\{i, j\} \in \Pi} \bar{P}_{ij}^\Lambda(Q_{ij}^*). \tag{16}$$

Since solving (16) entails comparison of at most $N(N - 1)/2$ quantities, the computational cost for obtaining the best $\Lambda$-pair is fairly low. Note that the minimizers in (16) may not be unique. Denote by $\Sigma$ the set of all the minimizers in (16) and by $|\Sigma|$ the number of elements in $\Sigma$. Two or more elements in $\Sigma$ are called equivalent $\Lambda$-pairs if they correspond to the same optimal buffer size as defined in (15). In other words, the equivalent $\Lambda$-pairs are the $\Lambda$-pairs that minimize (16) with same optimal buffer size. The following theorem summarizes the above results and gives a rigorous definition of the OPPTs.

**Theorem 4** Let $z(t)$ be a pure periodic trajectory with $\{i, j\}$ and $(0, \frac{Q}{r_i - r_y})$ as the switching sequence and switching instants in its first period, respectively. If $\{i, j\} \in \Sigma$ and $Q = Q^*_{ij}$ as defined in (15), then $z$ is an OPPT with period $T_{ij}$.

4.2 General Optimal Solutions

In this section, we will prove that the OPPT derived in the last section for the case $n_p = 1$ is actually an OS2 for an arbitrary $n_p$. Furthermore, if $\Sigma$ contains equivalent $\Lambda$-pairs, then the OPPT can be used as a building block to construct more complicated OS2s that are not pure. The main result of this section is the following theorem.

**Theorem 5** The OPPT defined in Theorem 4 is an OS2.

**PROOF.** Let $z(t)$ be an OPPT as defined in Theorem 4 and $P^*$ be its average power. Let $z(t) = (\sigma(t), \sigma(t))$ be an arbitrary periodic trajectory with average power $P$. We need to show that $P^* \leq P$. According to
Lemma 2, we can assume $z \in \Omega$. If $z$ is pure, then by the definition of $z^*$, we automatically have $P^* \leq P$. Hence, we assume that $z$ is mixed with $n_p > 1$ and its first period is as shown in Fig 8. Let $T_i$, $i = 0, \ldots, m$, be the successive time instants such that $q(T_i) = 0$. Denote by $P_i$, $i = 1, \ldots, m$, the average power of $C_{T_{i-1}, T_i}[z]$, which is the part of $z(t)$ within the interval $[T_{i-1}, T_i)$.

It is obvious that $P = P_1 T_1 + \cdots + P_m (T_m - T_{m-1})$ is a convex combination of $P_1, \ldots, P_m$. Thus $P \geq P^*$, where $i^* = \arg \min P_i$. Furthermore, we also have $P_{i^*} \geq P^*$, as otherwise the periodic extension of $C_{T_{i-1}, T_i}[z]$ is also pure but consumes less power than $z^*$, which contradicts the optimality of $z^*$. Hence, $P^* \leq P_{i^*} \leq P$.

When $|\Sigma| > 1$, the OPPT is not unique and neither is the OS2 according to Theorem 5. Furthermore, if $\Sigma$ contains equivalent $\Lambda$-pairs, we can use them to even construct an OS2 that is not pure. To see this, let $z_1$ and $z_2$ be two different OPPTs consisting of equivalent $\Lambda$-pairs with optimal buffer sizes $Q_1$ and $Q_2$, optimal period $T_1$ and $T_2$ and average power $P_1$ and $P_2$, respectively. By the definition of the equivalent $\Lambda$-pairs, we must have $P_1 = P_2$ and $Q_1 = Q_2$. Define $z_1 = C_{0, T_1}[z_1]$, $z_2 = C_{0, T_2}[z_2]$ and $z = P[J[z_1, z_2]]$. In other words, $z$ is a periodic trajectory with period defined by connecting one period of $z_1$ and $z_2$ together. It is obvious that $z$ consumes the same average power as $z_1$ and $z_2$. Thus $z$ is an OS2 with 2 different $\Lambda$-pairs, i.e., $n_p = 2$. In a similar way, more complicated OS2s can be constructed if $\Sigma$ contains more than two equivalent $\Lambda$-pairs.

Although mixed OS2s may exist, the OPPT is the simplest OS2 which can be easily computed and implemented. Thus we will focus on the OPPT in the rest of this paper for optimal solutions of Problem 2.

\subsection{5.3 $t_f$-adapted OPPT}

The OPPT is an infinite-length periodic trajectory that requires an infinite amount of incoming data. However, for real applications, the amount of data to be transferred is finite. Therefore, when the OPPT is used in a real application, another guard condition should be added to the system: switch component X to the lowest power mode (mode 1) whenever there is no more incoming data. Suppose that for an application, X needs to produce $t_f r_y$ amount of data for Y and this amount is not known during the design process. In this case, we can solve Problem 2 to obtain an OPPT $z_T = (q(t), \sigma(t))$ with period $T$. To evaluate how well $z_T$ works for this application, define the $t_f$-adapted trajectory, denoted by $A_{t_f}[z_T]$, as

$$A_{t_f}[z_T] = (\hat{q}(t), \hat{\sigma}(t)),$$

where

$$\hat{\sigma}(t) = \begin{cases} \sigma(t) & t \leq t_s \\ 1 & t_s < t \leq t_f, \end{cases}$$

and

$$\frac{d\hat{q}(t)}{dt} = \begin{cases} r_{\hat{\sigma}(t)} - r_y & t \leq t_s \\ -r_y & t_s < t \leq t_f, \end{cases}$$

where $t_s \neq t_f$ is the unique solution of the following equation

$$q(t_s) = (t_s - t_f)r_y.$$ 

In other words, $A_{t_f}[z_T]$ follows exactly the original trajectory $z_T$ up to the time $t_s$ when X finishes producing the $t_f r_y$ amount of data. During the interval $[t_s, t_f]$, component X is switched to the lowest power mode consuming a constant power $p_1$, while component Y is reading the remaining data in the buffer. Note that regardless of whether $r_1$ is 0 or not, X is not producing any new data during $[t_s, t_f]$ as all the $t_f r_y$ amount of data has been sent to the buffer before $t_s$. The $A_{t_f}[z_T]$ reflects what actually happens to the system when the strategy $z_T$ is applied to a real application with unknown but finite duration $t_f$. Although it comes from the optimal periodic trajectory $z_T$, it may not be optimal for this particular application unless $t_f$ is an integer multiple of $T$. However, without knowing $t_f$ \textit{a priori}, this is the best that can be achieved.

\section{5 Optimal Solutions for Fixed and Given $t_f$}

For unknown $t_f$, the OPPT is a good switching policy since it is the best periodic strategy that can be easily implemented in computer (resulting in a $t_f$-adapted trajectory). In this section, we study the case where the exact value of $t_f$ is known and derive optimal solutions to Problem 1 (OS1). An OS1 can be used to construct a periodic trajectory with period $t_f$ through the periodic extension. In this sense, Problem 1 can be thought of as a version of Problem 2 with fixed period $T = t_f$. With the constraint for the period, Problem 1 becomes more difficult than Problem 2. On the other hand, with the additional knowledge of $t_f$, we expect to obtain a solution that performs even better than the $t_f$-adapted OPPT for this given $t_f$.

Not surprisingly, the optimal solution to Problem 1 must also be a boundary-switching trajectory.
Lemma 3 If \( z \) is an OS1, then \( z \in \Omega \).

Remark 6 The perturbed trajectory \( z_h \) defined in (8) plays an important role in the proof of Lemma 2. However, since \( z_h \) has a different length from \( z \), it cannot be directly applied to prove Lemma 3 where the time horizon \( t_f \) is given and fixed. The key idea of the proof of Lemma 3 is to further perturb \( z_h \) using scaling operation with a proper parameter \( c \) so that \( S_c(z_h) \) has the same length as \( z \) and then show that the average power of \( S_c(z_h) \) is less than that of \( z \) for certain \( h \) if \( z \) has interior switchings. Refer to Appendix A for a complete proof.

Lemma 3 enable one to consider only the BSTs in finding the OS1s. Similar to the periodic case, in the rest of this section, we will first find a solution in a simple case where \( n_p = 1 \) and then prove that this solution is also an OS1 for an arbitrary \( n_p \).

5.1 Optimal Pure Trajectory (OPT)

The optimal pure trajectory (OPT) is defined as the optimal solution to Problem 1 under an additional constraint \( n_p = 1 \), i.e., only pure trajectories are considered as candidate solutions. As discussed in Section 3.5, any finite-length pure trajectory can be thought of as obtained from a \( \Lambda \)-trajectory through a folding operation. Thus the main task is to determine the best \( \Lambda \)-pair and the corresponding best folding parameter.

Let \( z_i(t) \) be a \( \Lambda \)-trajectory with length \( t_f \) and \( \Lambda \)-pair \( \{i,j\} \). Since \( t_f \) is fixed, its optimal buffer size is given by:

\[
Q_{ij} = \frac{t_f}{\alpha_{ij}}, \tag{18}
\]

where \( \alpha_{ij} \) is the constant defined in (12). Define \( z_m = F_{m\{z_i\}} \). Then \( z_m \) is a pure trajectory as shown in Fig. 9 with \( 2m \) switchings and the same \( \Lambda \)-pair as \( z_i \). As discussed in Section 3.5, \( z_m \) has an optimal buffer size \( Q_{ij}/m \) and its average power is

\[
\bar{P}_{ij}^T(m) = \beta_{ij} + \frac{2mk_4}{t_f} + \frac{p_{ij}t_f}{m\alpha_{ij}}, \tag{19}
\]

where \( \beta_{ij} \) is the constant defined in (13). Taking the derivative of \( \bar{P}_{ij}^T \) with respect to \( m \) and setting it to zero, we obtain the optimal value of \( m \) as

\[
m_{ij}^* = t_f \sqrt{\frac{p_h}{2k_4\alpha_{ij}}} \tag{20}
\]

Note that the folding parameter \( m \) must be an integer, and the function \( \bar{P}_{ij}^T(m) \) is convex in \( m \). Therefore, if \( m_{ij}^* \) is not an integer, the optimal feasible value of \( m_{ij} \) is \( m_{ij}^* \cdot m \), is whichever of the two neighboring integers of \( m_{ij} \) that results in a smaller value of \( P_{ij}^T(m) \) as defined in (19).

Hence,

\[
m_{ij}^* = \arg\min_{m \in \{\lfloor m_{ij}^* \rfloor, \lceil m_{ij}^* \rceil \}} P_{ij}^T(m). \tag{21}
\]

The minimal achievable power with the \( \Lambda \)-pair \( \{i,j\} \) is \( P_{ij}^T(m_{ij}^*) \). Then the best \( \Lambda \)-pair \( \{\sigma_{ij}^+, \sigma_{ij}^-\} \) can be obtained as

\[
\{\sigma_{ij}^+, \sigma_{ij}^-\} = \arg\min_{\{i,j\} \in \Sigma_f} P_{ij}^T(m_{ij}^*). \tag{22}
\]

Denote by \( \Sigma_f \) the set of all minimizers of (22) and by \( |\Sigma_f| \) the number of elements in \( \Sigma_f \). The following theorem summarizes the above results.

Theorem 7 Let \( z_m(t) \) be a pure trajectory as shown in the right side of Fig. 9 with \( 2m \) switchings and \( \Lambda \)-pair \( \{i,j\} \). If \( \{i,j\} \in \Sigma_f \) and \( m = m_{ij}^* \), then \( z_m \) is an OPT with optimal buffer size \( Q_{ij}/m_{ij}^* \).

5.2 General Optimal Solution

In last subsection, we derive analytically the optimal pure trajectories with \( n_p = 1 \). A natural question is that whether the power can be further reduced if we relax the constraint on \( n_p \). To answer this question, we start with a simple case where the candidate trajectories are allowed to contain at most two distinct \( \Lambda \)-pairs,\(^4\) i.e., \( n_p \leq 2 \). Let \( z_{m_1, m_2} \) be a BST consisting of \( m_1 \) copies of \( \Lambda \)-pair \( \{i_1, j_1\} \) and \( m_2 \) copies of \( \Lambda \)-pair \( \{i_2, j_2\} \). Without loss of generality, assume that all the same pairs are grouped together as shown in Fig 10. In other words, the switching sequence of \( z_{m_1, m_2} \) is assumed to take the following form

\[
(\sigma_1, \ldots, \sigma_{2(m_1+m_2)}) = (i_1, j_1, \ldots, i_1, j_1, i_2, j_2, \ldots, i_2, j_2). \tag{23}
\]

(\( m_1 \) pairs \( m_2 \) pairs)

\(^4\) Two different \( \Lambda \)-pairs may consist of three or four different modes. For example, \( \{\sigma_1, \sigma_2\} \) and \( \{\sigma_1, \sigma_3\} \) are also called two different \( \Lambda \)-pairs although they have one mode in common.
The optimal buffer size of $z_{m_1,m_2}$ is uniquely determined by

$$Q = \frac{t_f}{\alpha_{i,j,m_1} + \alpha_{i,j,m_2}},$$

where $\alpha_{i,j}$ is the constant defined in (12). Let $\beta_{i,j}$ be the running power of the $\Lambda$-pair $(i,j)$ as defined in (13). Then the total energy consumed by $z_{m_1,m_2}$ is computed as

$$E(m_1,m_2) = 2(m_1 + m_2)k_\delta + p_\delta t_f^2 + \beta_{i,j} \alpha_{i,j,m_1}m_1 t_f + \beta_{i,j} \alpha_{i,j,m_2}m_2 t_f.$$

**Lemma 4** For any $\{i_1,j_1\}$ and $\{i_2,j_2\}$, there is a pair of nonnegative integers $(m_1, m_2)$ with either $m_1^* = 0$ or $m_2^* = 0$ such that $E(m_1, m_2) \leq E(m_1^*, m_2)$, for any other pair of nonnegative integers $(m_1^*, m_2)$.

**Proof.** For simplicity, let $a_1 = \beta_{i,j} \alpha_{i,j,m_1}, a_2 = \beta_{i,j} \alpha_{i,j,m_2}, c = p_\delta t_f^2$. Relax $m_1, m_2$ to nonnegative real numbers $x_1$ and $x_2$. Then

$$E(x_1, x_2) = 2k_\delta(x_1 + x_2) + a_1 x_1 + a_2 x_2 + c \frac{a_1 x_1 + a_2 x_2 + c}{\alpha_{i,j,m_1} x_1 + \alpha_{i,j,m_2} x_2}.$$

Note that all the constants $a_1, a_2, c, \alpha_{i,j,m_1}$, and $\alpha_{i,j,m_2}$ are positive. To prove the lemma, it suffices to show that there exists a point on the $x_1$ or $x_2$ axis that minimizes $E(x_1, x_2)$ in the first quadrant. To find the minimizers of $E(x_1, x_2)$ in the first quadrant, we can first minimize it along each ray in the first quadrant, and then find the ray that gives the best minimum value. Towards this purpose, define $x_2 = \lambda x_1$, where $\lambda \in [0, \infty)$. Then

$$E(x_1, \lambda x_1) = 2k_\delta(1 + \lambda) x_1 + \frac{(a_1 + a_2 \lambda) x_1 + c}{(\alpha_{i,j,m_1} + \alpha_{i,j,m_2}) x_1}.$$

Thus $E(x_1, \lambda x_1)$ is the minimum value achieved on the ray $x_2 = \lambda x_1$. To prove the lemma, it suffices to show that either $\lambda = 0$ or $\lambda = \infty$ minimizes $E(x_1, \lambda x_1)$. After some computations, $E(x_1, \lambda x_1)$ reduces to

$$E(x_1, \lambda x_1) = d_3 \sqrt{d_2 y + \frac{1}{\alpha_{i,j,m_2}}} + d_4 y + \frac{a_2}{\alpha_{i,j,m_2}} \Delta f(y),$$

where $d_1 = a_1 \alpha_{i,j,m_2} - a_2 \alpha_{i,j,m_1}$, $d_2 = \lambda x_1 - \alpha_{i,j,m_1}$, and $d_3 = 2\sqrt{2k_\delta} \Delta c$ are all constants and

$$y = \frac{1}{\alpha_{i,j,m_2} \lambda + \alpha_{i,j,m_1} \alpha_{i,j,m_2}}.$$

Note that except $d_1$ and $d_2$, all the other constants are positive. As $\lambda$ increases from 0 to $\infty$, $y$ decreases from $\frac{1}{\alpha_{i,j,m_1} \alpha_{i,j,m_2}}$ to 0. Hence, it suffices to show that either 0 or $\frac{1}{\alpha_{i,j,m_1} \alpha_{i,j,m_2}}$ is a minimizer of $f(y)$ in $[0, \frac{1}{\alpha_{i,j,m_1} \alpha_{i,j,m_2}}]$. Note that the second-order derivative of $f(y)$ is

$$\frac{d^2 f}{dy^2}(y) = -\frac{d^2 f}{dy^2} \leq 0.$$

Thus $f(y)$ is a concave function of $y$ in $[0, \frac{1}{\alpha_{i,j,m_1} \alpha_{i,j,m_2}}]$. Since the minimizer of a concave function over a bounded set must be on the boundary of the set, we conclude that either 0 or $\frac{1}{\alpha_{i,j,m_1} \alpha_{i,j,m_2}}$ is a minimizer of $f(y)$ in $[0, \frac{1}{\alpha_{i,j,m_1} \alpha_{i,j,m_2}}]$.

According to Lemma 4, for any given two $\Lambda$-pairs, we can always use one of them to construct a pure trajectory that performance equally well or better than all the other mixed trajectories involving these two $\Lambda$-pairs. Therefore, the following corollary follows immediately.

**Corollary 1** The OPT is an optimal solution to Problem 1 under an additional constraint $n_p \leq 2$.

The question now becomes whether more energy can be saved by further relaxing the constraint on $n_p$. It turns out to be not the case. In fact, the OPT is an optimal solution to Problem 1 for an arbitrary $n_p$. This can be proved by induction. The following lemma is the key of the induction procedure.

**Lemma 5** For any BST $z$ with length $t_f$ and $n_p = l + 1$, there exists another BST $\hat{z}$ with length $t_f$ and $n_p \leq l$ that consumes equal or less power than $z$.

The proof Lemma 5 can be found in Appendix B. By this lemma, any BST corresponds to a pure trajectory with no more power consumption. Thus the following theorem follows immediately.
Theorem 8 The OPT defined in Theorem 7 is an OS2 for an arbitrary $n_p$.

5.3 OPT vs. $(t_f$-adapted) OPPT

In this subsection, we compare the OPT derived in last subsection with the $(t_f$-adapted) OPPT derived in Section 4 to get a clearer picture of how the corresponding two problems, initially formulated from two different practical aspects, relate to each other.

Proposition 1 Let $(z_T)$ be an OPPT with period $T$ and $z_{t_f}$ be an OPT with length $t_f$ and $2m$ switchings. Denote by $(z_{t_f}^T)$ the $t_f$-adapted trajectory of $(z_T)$. Let $P(\cdot)$ be the average power for a given trajectory. Then the three trajectories relate to each other in the following aspects:

1. $P(z_T) \leq P(z_{t_f}) \leq P(z_{t_f}^T)$ for any $t_f \geq 0$;
2. If $t_f = l \cdot T$ for some $l \in \mathbb{N}$, then $P(z_{t_f}) = P(z_T) = P(z_{t_f}^T)$;
3. $P(z_{t_f}) \rightarrow P(z_T)$ and $P(z_{t_f}^T) \rightarrow P(z_T)$ as $t_f \rightarrow \infty$.

PROOF. (i) Obviously $P(z_{t_f}) \leq P(z_{t_f}^T)$ as $(z_{t_f}^T)$ is also a trajectory with length $t_f$. Since $z_T$ is best periodic trajectory, $P(z_T) \leq P(C_0, t_f)[z_T] = P(z_T)$. The desired result follows. 2. Since $t_f = l \cdot T$, then $C_0, t_f[z_T](t_f) = 0$. Thus $C_0, t_f[z_T]$ is also a feasible trajectory in $[0, t_f]$. By the optimality of $z_T$, we have $P(z_{t_f}) \leq P(C_0, t_f[z_T]) = P(z_T)$. Considering the result in (ii), we have $P(z_{t_f}) = P(z_T)$. According to (17), $z_{t_f}^T = z_{t_f}$ when $t_f = l \cdot T$. Thus all the three trajectories consume the same power in this case. (iii) Considering the result in 1), it suffices to prove that $P(z_{t_f}^T) \rightarrow P(z_T)$ as $t_f \rightarrow \infty$. Let $m_t = \lfloor t_f/T \rfloor$. Then obviously, $m_t T/t_f \rightarrow 1$ as $t_f \rightarrow \infty$. Denote by $P_1$ and $P_2$ the average power of $(z_{t_f}^T)$ during the interval $[0, m_t T]$ and $[m_t T, t_f]$, respectively. According to (17), $t_s > m_t T$ and $z_{t_f}^T(t) = z_T(t)$ for $t \in [0, m_t T]$. Thus $P_1 = P(z_T)$. Note that $P_2 \leq p_N + \frac{2k_s}{m_t T} + p_b Q$, where $p_N$ is the power of the highest mode. Hence,

$$P(z_{t_f}^T) = \frac{P_1 m_t T + P_2(t_f - m_t T)}{t_f} \xrightarrow{t_f \to \infty} P(z_T).$$

Remark 9 Any practical application corresponds to a finite $t_f$. If the $t_f$ is unknown, we can only compute the OPPT $(z_T)$. Applying $(z_T)$ to the application results in a $t_f$-adapted trajectory $(z_{t_f}^T)$. On the other hand, if $t_f$ is known a priori, a better trajectory $(z_{t_f})$ than $(z_{t_f}^T)$ can be computed. In fact, $(z_T)$ is the best trajectory for the given $t_f$ and its power is bounded from below by $P(z_T)$.

6 Simulation

6.1 Fictional Example

Consider a system (H1) with 6 power modes as defined in Table 1. Assume that $k_s = 0.1$, $p_b = 0.1$ and $r_s = 3.5$. For this system H1, compute the OPPT $(z_T)$, the OPT $(z_{t_f})$, and the $(t_f$-adapted) OPPT $(z_{t_f}^T)$, according to Theorem 4, Theorem 8 and equation (17), respectively. Denote by $\bar{P}()$ the average power of a given trajectory. In Fig. 11-(a), we plot the power of each trajectory as a function of $t_f$. It can be seen that $\bar{P}(z_{t_f})$ always stays below $\bar{P}(z_{t_f}^T)$ and both of them converge to $\bar{P}(z_T)$ from above as $t_f \rightarrow \infty$. Also observed is that the three trajectories have the same average power when $t_f$ is an integer multiple of the optimal period $(T = 2.8284)$ of the OPPT. These observations are consistent with our analysis in Section 5.3. In Fig. 11-(b), we plot the optimal A-pairs of $z_T$ and $z_{t_f}$ as functions of $t_f$. It can be seen that the optimal A-pair in $z_{t_f}$ is initially $(5, 2)$ and eventually converges to $(5, 4)$ which is the optimal A-pair of $z_T$ (and $z_{t_f}^T$). This indicates that $z_{t_f}$ may involve different A-pairs for different $t_f$.

6.2 Practical Example

Our theoretical results can be applied in many real-world applications, such as the power management problem of a multiple-speed disk Gurumurthi et al. [2003] and the dynamic voltage scheduling (DVS) problem of a variable speed processor Burd and Brodersen [2000]. In this section, we use a DVS example to illustrate the effectiveness of our results.

Table 1

<table>
<thead>
<tr>
<th>mode</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>$r_s$</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>$p_b$</td>
<td>0.1</td>
<td>0.12</td>
<td>0.2</td>
<td>0.3</td>
<td>0.33</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$f_s$ (MHz)</th>
<th>150</th>
<th>400</th>
<th>600</th>
<th>800</th>
<th>1000</th>
</tr>
</thead>
<tbody>
<tr>
<td>$r_s$ (MB/s)</td>
<td>0.45</td>
<td>1.2</td>
<td>1.8</td>
<td>2.4</td>
<td>3.0</td>
</tr>
<tr>
<td>$p_b$ (Watt)</td>
<td>0.08</td>
<td>0.17</td>
<td>0.4</td>
<td>0.9</td>
<td>1.6</td>
</tr>
</tbody>
</table>

Let X be an Intel Xscal processor Xu et al. [2004] with five available power modes as defined in Table 2. Suppose that Y is a video card that fetches data from X at a constant speed 8 Mbps (1MB/s). The power per megabyte for buffer B can be looked up in the datasheet Inc. [2000] as $6.258 \times 10^{-4}$ W/MB. A typical value of the switching energy is 0.1 mJ in a microprocessor Burd and Brodersen [2000]. Since the switching cost $k_s$ in our model may also
include other switching penalties, such as the switching delay penalty, we will test our method for \(k_s\) ranging from 0.1mJ to 100mJ. As \(t_f\) is usually large for video programs, the \(t_f\)-adapted OPPT and the OPT will provide almost the same power performance. For simplicity we only implement the \(t_f\)-adapted OPPT in this simulation and refer to this method as Scheme 1. A heuristic strategy, referred to as Scheme 2, is also implemented where \(X\) is switched to the highest speed until the buffer is full and then switched to the lowest speed until the buffer is empty. Scheme 2 is tested for four heuristically selected buffer sizes 0.1MB, 0.3MB, 1MB and 8MB. The power consumptions of Scheme 2 in these cases are compared with Scheme 1 in Fig. 12. It is clear that the proposed optimal strategy always performs the best for each \(k_s\) and can save about 60% of power consumption compared with the heuristic ones.

7 Conclusions

In this paper, we introduce a modeling framework for the dynamic power management problem using hybrid systems. The power minimization problem is formulated as a class of optimal control problems. By exploiting the particular structure of the system model, the optimal solutions are derived analytically based on some variational approach. Simulation result based on real data shows that the proposed method can save 60% energies compared with a heuristic scheme. Future research will focus on the following two aspects: one is to extend our analysis to the case where more than one buffers are inserted among multiple streamlined components. The other one is to study the case where the data rates of components are varying or even random instead of constant.

A Proof of Lemma 3

PROOF. Let \(z(t) = (q(t), \sigma(t))\) be an OS1 with switching instants \((t_0, \ldots, t_{n-1})\) and buffer size \(Q\). Suppose \(q(t_i) \in (0, Q)\) for some \(i\). Define \(z_h\) in (6). Then \(z_h\) has the same buffer size \(Q\) as \(z\), and similar to (9), its length is \(t_f^h = t_{i-1} + t_h^{(2)} + (t_f - t_i + 1)\), where \(t_h^{(2)}\) is given in (7). Define \(\hat{z}_h = S_{c_h}[z_h]\), where \(c_h = t_f/t_h\). According to the properties of the scaling operation, the buffer size of \(\hat{z}_h\) becomes \(cQ\) and the length of \(\hat{z}_h\) is changed back to \(t_f\). Therefore, \(\hat{z}_h\) is a feasible trajectory for Problem 1. Considering (3) and (10), the power of \(\hat{z}_h\) is computed as

\[
\hat{P}(\hat{z}_h; Q, t_f) = \frac{1}{t_f} \left[ E_1 + E_3 + 2k_s + p_{\sigma_h} \cdot h + p_{\sigma_{i+1}} (t_h^{(2)} - h) + p_b c_h Q + \frac{n k_s (1 - c_h)}{c_h t_f} \right].
\]
Taking the derivative of \( \dot{P}(z_h; Q, t_f) \) with respect to \( h \), we have
\[
\frac{dP(z_h; Q, t_f)}{dh} = \frac{1}{\left( \frac{b}{t_f} \right)^2} \left[ \left( \frac{p_{\sigma_1} + p_{\sigma_1+1} r_y - r_{\sigma_1}}{r_{\sigma_1+1} - r_y} \right) \cdot \left( t_f - t_{i+1} - t_{i-1} \right) + \frac{q_2 - q_1}{r_{\sigma_1+1} - r_y} \right] - \left( 1 + \frac{r_y - r_{\sigma_1}}{r_{\sigma_1+1} - r_y} \right) \cdot \left( E_1 + E_3 + k_s + p_{\sigma_1+1} \right) \cdot \frac{q_2 - q_1}{r_{\sigma_1+1} - r_y} \right] + \frac{n_k_s - p_n Q t_f}{(t_f)^2} \left( 1 - \frac{r_y - r_{\sigma_1}}{r_{\sigma_1+1} - r_y} \right).
\]

Therefore, \( \dot{P}(z_h; Q, t_f) \) is monotone with respect to \( h \) as the sign of \( \frac{dP(z_h; Q, t_f)}{dh} \) does not depend on \( h \). Using the same argument as in the last paragraph of the proof of Lemma 2, it follows that the optimal solution to Problem 1 must also be a BST.

\section*{B \hspace{1cm} Proof of Lemma 5}

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{trajectory}
\caption{A trajectory \( z \) with \( n_p = l + 1 \)}
\end{figure}

\textbf{PROOF.} Without loss of generality, assume that \( z \) is as shown in Fig. B.1 with the following switching sequence
\[
(i_1, j_1, \ldots, i_m, j_m, \ldots, i_{m+1}, j_{m+1}).
\]

Define \( z_1 = C_{0,1}(z) \) and \( z_2 = C_{a,1}(z) \), where \( a \) is the starting time of the first copy of \( \{i_2, j_2\} \) in Fig. B.1. Let \( \{\sigma_+, \sigma_-\} \) be a \( \Lambda \)-pair whose data rates and powers are defined as
\[
r_{\sigma_+} = r_y + \frac{2Q}{t_f}, \quad r_{\sigma_-} = r_y - \frac{2Q}{t_f},
\]
\[
p_{\sigma_+} = \frac{\sum_{k=2}^{l+1} p_{i_k} (\tau_{i_k})}{\tau_f - a}, \quad p_{\sigma_-} = \frac{\sum_{k=2}^{l+1} p_{j_k} (\tau_{j_k})}{\tau_f - a}, \quad \text{(B.1)}
\]

where \( \tau_{i_k} \) is the time \( z_2 \) spends in mode \( i_k \). Thus \( p_{\sigma_+} \) and \( p_{\sigma_-} \), in the above equation represent the average powers of all the ascending modes and descending modes of \( z_2 \), respectively. Note that the so defined modes \( \sigma_+ \) and \( \sigma_- \) may not be in \( S \) and even may not be physically meaningful. They are only introduced to simplify our discussion.

Let \( \bar{z} \) be a \( \Lambda \)-trajectory with the virtual modes \( \{\sigma_+, \sigma_-\} \) as shown in Fig. B.2-(a). Define the switching cost in \( \bar{z} \) to be \( 2lk_{2}\). Then \( \bar{z} \) has the same buffer size and total energy as \( z_2 \); thus \( J[\bar{z}, \hat{z}] \) consumes the same energy as \( z \). On the other hand, \( J[z_1, \hat{z}] \) is a mixed trajectory with \( n_p = 2 \); its two different \( \Lambda \)-pairs \( \{\sigma_+, \sigma_-\} \) and \( \{i_1, j_1\} \) appear \( m_1 \) times, respectively. By Lemma 4, there exists a pure trajectory \( \hat{z} \) involving only the pair \( \{i_1, j_1\} \) or \( \{\sigma_+, \sigma_-\} \) that consumes no more energy than \( J[z_1, \hat{z}] \). Thus \( \hat{z} \) also consumes no more energy than \( z \). If \( \hat{z} \) involves only the pair \( \{i_1, j_1\} \), then it is a pure trajectory \( (n_p = 1) \) satisfying all the constraints in Problem 1 with no more energy than \( z \). On the other hand, if \( \hat{z} \) involves only the pair \( \{\sigma_+, \sigma_-\} \), it may not satisfy the third constraint in Problem 1 as \( \sigma_+ \) and \( \sigma_- \) may not be in \( S \). In this case, \( \hat{z} \) must consist of a series of a scaled version of \( z_2 \), i.e., \( z = J_m[S_c[z_2]] \) for some \( m \) and \( c \). According to (B.1), \( J_m[S_c[z_2]] \) (obtained by replacing every \( z_2 \) with \( 2z_2 \) as shown in Fig B.2-(b) consumes the same energy as \( \hat{z} \). Thus we obtain an trajectory \( J_m[S_c[z_2]] \) with \( n_p = 1 \) (involves only the valid modes \( i_2, j_2, \ldots, i_{t+1}, j_{t+1} \)) that consumes no more energy than \( z \). Hence, in either case we can find a trajectory with \( n_p \leq l \) that satisfies all the constraints in Problem 1 and consumes no more power than \( z \).

\section*{References}


Note that \( \{\sigma_+, \sigma_-\} \) and \( \{i_1, j_1\} \) have different switching costs. However, with few changes, Lemma 4 also works for the case where the two \( \Lambda \)-pairs have different switching costs.
Fig. B.2. (a) Two-mode representation of $z_2$; (b) The Optimal trajectory $\hat{z}$ and its corresponding original trajectory


