

# 动态电源管理的在线优化技术

翟剑锋, 李衍杰<sup>†</sup>, 陈浩耀

(哈尔滨工业大学 深圳研究生院, 广东 深圳 518055)

**摘要:** 研究离散事件动态系统中的一类随机离散动态系统——半Markov决策过程, 在动态电源管理问题中的应用. 动态电源管理问题存在于很多便携式电子设备中, 其主要目的是根据电子设备的状态通过电源管理策略选择关闭或休眠一些元器件, 从而实现节省电子设备功耗, 延长电池使用时间的目的. 首先讨论了动态电源管理问题的建模, 给出了一种带有禁止时间的在线优化方法, 该方法通过设备自身运行数据, 自主地学习并改进电源的动态管理策略, 从而使每台电子设备具有个性化的动态电源管理方式, 其优化过程可以在设备充电时完成, 不需要通过云传输和云计算, 避免了隐私数据的泄漏. 最后通过仿真实验验证了算法的有效性.

**关键词:** 动态电源管理; 智能节能; 半Markov决策过程; 在线优化

**引用格式:** 翟剑锋, 李衍杰, 陈浩耀. 动态电源管理的在线优化技术. 控制理论与应用, 2018, 35(1): 65 – 70

**中图分类号:** TP273      **文献标识码:** A

## Online optimization of dynamic power management

ZHAI Jian-feng, LI Yan-jie<sup>†</sup>, CHEN Hao-yao

(Shenzhen Graduate School, Harbin Institute of Technology, Shenzhen Guangdong 518055, China)

**Abstract:** This paper mainly considers the application of Semi-Markov decision process (SMDP), which is a stochastic discrete event dynamic system (DEDS), in the dynamic power management (DPM) problems. There are the DPM problems in many portable devices, which can provide satisfactory performance with minimum power consumption by selectively turning off system components, which enter the idle state, to save the energy consumption and prolong the service time. Firstly, the SMDP model of DPM is introduced and then an online optimization approach with forbidden time is presented. This approach can learn to improve the policy of DPM by analyzing the data from the device itself and make the device have an individual power management policy. The optimization can be implemented during the charging and thus does not need the cloud transportation and computation, which avoids the leak of private data. Finally, a simulation example is applied to illustrate the applicability of the approach.

**Key words:** dynamic power management; intelligent energy saving; semi-Markov decision process; online optimization

**Citation:** ZHAI Jianfeng, LI Yanjie, CHEN Haoyao. Online optimization of dynamic power management. *Control Theory & Applications*, 2018, 35(1): 65 – 70

## 1 引言(Introduction)

随着互联网和通讯业的迅速发展, 智能手机、笔记本等便携式电子设备已广泛融入到日常生活中. 电子设备的功能愈来愈丰富, 智能化程度越来越高, 其对移动电能的需求也愈来愈迫切, 然而电池的容量却增长较为缓慢, 很难满足电子设备的需求. 动态电源管理通过有选择地将进入空闲状态的系统组件关闭或休眠的方式来最小化系统的功耗<sup>[1-2]</sup>, 从而达到不增加任何硬件情况下节能的效果.

动态电源管理策略一般可分为3类: 超时策略、预测性策略<sup>[3]</sup>和基于随机模型的电源管理策略. 相比较而言, 超时策略和预测性策略通常情况下是启发式的,

主要依赖于经验设置, 而基于随机模型的电源管理策略则能够提供最优的电源管理策略. 基于模型的电源管理早期主要集中于离散时间策略<sup>[4-5]</sup>, 以离散时间Markov决策过程(discrete-time Markov decision process, DTMDP)为主要的数学模型. 文献[6]利用该模型研究了集群系统的动态电源管理问题. 离散时间策略在每一个时刻需要判断是否采用节能行动, 即使状态未发生改变也进行判断, 因而需要消耗额外的能耗. 为了避免这类问题, 连续时间Markov决策过程(continuous-time Markov decision process, CTMDP)被应用于动态电源管理问题<sup>[7]</sup>. 该模型是事件驱动的, 即状态发生改变后才考虑是否采取节能行动, 因而避免了

收稿日期: 2016-11-16; 录用日期: 2017-10-26.

<sup>†</sup>通信作者. E-mail: autolyj@hit.edu.cn; Tel.: +86 755-26033788.

本文责任编辑: 赵千川.

国家自然科学基金项目(61004036), 深圳市基础计划项目(JCYJ20150731105106111)资助.

Supported by the National Natural Science Foundation of China (61004036) and Shenzhen Basic Research Program (JCYJ20150731105106111).

离散时间策略的不足. 然而CTMDP假设系统在任何状态的逗留时间都服从指数分布, 这在实际情形下是一种理想化假设, 很多情形不满足该假设. 半Markov决策过程(SMDP)模型可以描述服从任意时间分布的系统, 避免了指数分布假设所带了的限制<sup>[8]</sup>, 因而可以在动态电源管理问题中得到应用. 一般的SMDP模型没有考虑系统组件在空闲状态已停留了多少时间. 基于日常经验, 在空闲状态已停留的时间对于设计电源管理策略具有重要的影响. 为此, 文献[9]提出了时间索引的半Markov决策过程模型(time-indexed semi-Markov decision process, TISMDP), 该模型把处于空闲状态的时间分成一系列的时间片段, 这些片段将空闲状态进行了细分, 可以记录已停留在空闲状态的时间. 基于TISMDP模型, 文献[9]得出了令人满意的结果. 然而, 该方法依赖于精确的随机模型参数, 不精确的模型将会对节能效果产生较大的影响. 此外, 模型的参数统计比较复杂, 需要较大的计算量, 因而该方法需要离线实现. 在在线优化方面, 文献[10]基于半Markov决策过程给出了一种基于等价无穷小描述的在线优化算法, 无穷小矩阵的计算增加了一定的运算量. 文献[11]综述了随机动态电源管理在无线传感网络中的研究现状, 并指出动态电源管理问题随着模型参数的增加求解变得更加复杂.

便携式设备的用户往往是不同的, 每个用户都具有不同的使用习惯, 结合用户的使用习惯来优化设备的电源管理策略具有很好的应用价值. 本文提出了一种带有禁止时间的在线优化方法, 该方法通过采集设备自身运行数据, 自主地学习并改进电源的动态管理策略, 从而使每台电子设备具有个性化的动态电源管理策略, 其优化过程可以集中在设备充电时完成, 因而不需要通过云传输和计算, 避免了隐私数据的泄漏. 由于该方法仅需要系统的运行轨迹, 不依赖于模型参数, 因而也避免了模型参数对优化效果的影响. 该方法的主要思想在会议版论文[12]中介绍过, 在本文中引入了禁止时间的思想. 由于在线优化方法利用随机梯度进行寻优, 可能会收敛到局部最优策略, 导致设备在较早时间内切换到休眠状态, 增加错误的休眠和不必要的能耗, 为此在模型中引入禁止时间, 在禁止时间内禁止设备休眠, 从而可避免设备错误地在较早时间内切换到休眠状态, 在此基础上再利用在线优化方法在禁止时间之外的时间内进行优化, 从而实现降低能耗和降低错误休眠率的目的. 最后, 通过仿真测试验证了算法的有效性.

## 2 动态电源管理问题(Dynamic power management)

假设设备有3种状态: 工作状态、空闲状态和休眠状态(低功耗状态). 当请求队列中没有用户请求且设备处于空闲状态时, 系统状态将会从工作状态转移到

空闲状态. 此时, 系统根据电源管理策略选择是否切换进入休眠状态. 如果选择进入休眠状态, 设备转入低功耗运行. 如果选择不切换, 则设备将继续高能耗运行. 当请求队列中有请求到达时, 设备将被唤醒返回到工作状态. 设备进入休眠状态和唤醒返回工作状态时都需要消耗能量.

为了达到节约能耗的目的, 空闲时间需要足够长才能保证节能. 定义最小的空闲时间 $T_{be}$ 为系统选择切换和不切换所消耗的功耗达到平衡的时间, 那么它满足下式

$$P_{ow}(\text{idle}) \times T_{be} = E_{sd} + E_{wu} + P_{ow}(\text{sleep}) \times (T_{be} - T_{sd} - T_{wu}), \quad (1)$$

其中 $P_{ow}(\text{idle})$ 和 $P_{ow}(\text{sleep})$ 分别表示在空闲状态和休眠状态的能耗率;  $E_{sd}$ 和 $E_{wu}$ 分别表示每次休眠和唤醒所需的能量;  $T_{sd}$ 和 $T_{wu}$ 分别表示休眠和唤醒所需要的时间. 令 $T_{ms}$ 为最小的休眠时间, 则

$$T_{ms} = T_{be} - T_{sd} - T_{wu}. \quad (2)$$

如果休眠时间比 $T_{ms}$ 短, 休眠将是错误的行动, 不仅不能节约电量反而会消耗更多电量. 定义错误休眠率为

$$R_{wd} = \frac{N_{wd}}{N_{sd}}, \quad (3)$$

其中:  $N_{wd}$ 表示错误休眠次数,  $N_{sd}$ 表示所有休眠次数. 为避免设备错误地在较早的时间内切换到休眠状态, 造成能耗的浪费, 在本文中, 引入了一个禁止时间 $T$ . 当系统处于空闲状态且在空闲状态的逗留时间小于 $T$ 时, 禁止设备进入休眠状态.

用户请求可以通过请求到达时间间隔分布来描述, 文献[9]研究表明, 当系统处于工作状态时, 请求到达的时间间隔可用指数分布来描述, 即请求时间间隔的分布函数可描述为

$$F_1(t) = 1 - e^{-\lambda t}, \quad (4)$$

其中 $\lambda$ 是请求到达时间间隔率. 当系统处于空闲状态时, 请求时间间隔的分布可用Pareto分布描述, 即此时的分布函数可描述为

$$F_2(t) = 1 - at^{-b}, \quad (5)$$

其中 $a$ 和 $b$ 是分布函数的参数. 此外, 设备的服务时间可以假设服从指数分布.

在本文中, 在TISMDP模型<sup>[9]</sup>基础上引入了禁止时间, 带有禁止时间的TISMDP模型可用图1描述. 在图1中, 当设备处于禁止时间状态时, 设备仍采用空闲状态功率运行, 禁止设备进入休眠状态. 等设备空闲时间超过禁止时间 $T$ 后, 设备何时进入休眠状态由电源管理策略决定. 为了考虑设备已空闲的时间, 空闲状态和休眠状态被分别用时间片段 $\Delta t$ 分为 $M$ 个状态, 其中 $U$ 代表休眠时间, 加上禁止时间、工作和唤醒3个状态, 整个系统系统可以用 $2M + 3$ 个状态描述, 状态

空间包括禁止时间(dtime)、工作状态(active)、唤醒状态(wake)、 $M$  个空闲状态(idle)和  $M$  个休眠状态(sleep). 将状态空间记为  $S = \{1, 2, \dots, 2M + 3\}$ . 令  $\tau_n$  表示状态转移时刻,  $n = 0, 1, 2, \dots$ . 系统在空闲状态有两个动作: 休眠和不休眠, 分别用  $a_1$  和  $a_2$  表示. 因而, 在空闲状态的行动集为  $A = \{a_1, a_2\}$ , 而在其它状态则不需要采取任何行动, 也可以理解为只有一个行动. 令  $(X_n, A_n)$  表示在每个时刻  $\tau_n$  的状态行动对. 在时刻  $\tau_n$  从状态  $i \in S$  到状态  $j \in S$  的转移概率可以描述为

$$p(j|i, a) = P(X_{n+1} = j | X_n = i, A_n = a). \quad (6)$$

令  $F(t|i, a, j)$  表示系统执行动作  $a$  时, 从状态  $i$  到状态  $j$  的逗留时间分布, 那么, 系统在状态  $i$  执行动作  $a$  时的期望逗留时间为

$$\tau(i, a) = \sum_{j \in S} p(j|i, a) \int_0^\infty tF(dt|i, a, j), \quad (7)$$

为了得到上述的模型参数, 文献[9]给出了复杂详细的分析和计算.

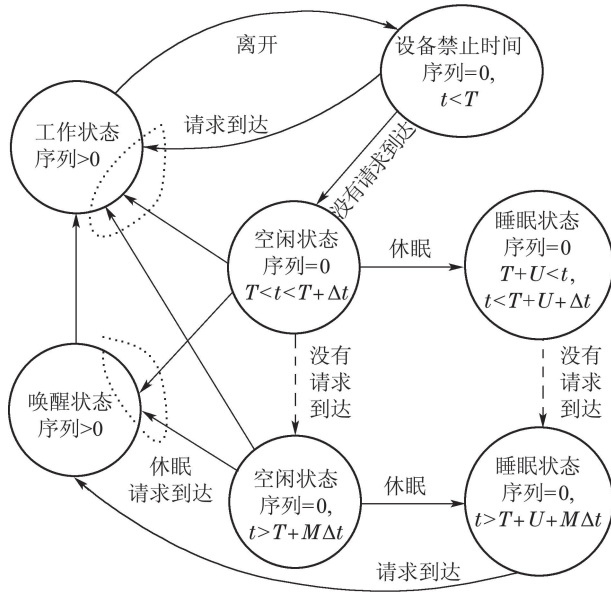


图 1 带有禁止时间的时间索引半Markov决策模型

Fig. 1 Time indexed semi-Markov decision model with forbidden time

令  $f(X_n, A_n)$  表示系统在  $\tau_n$  时刻的即时能耗. 当设备休眠或唤醒时有即时能耗, 因此有

$$f(\text{idle}, a_1) = E_{sd}.$$

由于唤醒状态不需要采取行动, 其即时功耗为

$$f(\text{wake}) = E_{wu}.$$

其他状态的即时功耗为0. 考虑到在不同状态下的持续功耗, 可以得到时刻  $\tau_n$  和  $\tau_{n+1}$  之间的期望总功耗为

$$\text{POW}(i, a) = E\{[f(X_n, A_n) + P_{ow}(X_n)(\tau_{n+1} - \tau_n)] | X_n = i, A_n = a\}.$$

在空闲状态时, 考虑随机型策略  $u$ , 即在系统状态为  $i$  时以概率  $u(a|i)$  选择动作  $a$ . 在该策略下, 状态转移概率, 期望逗留时间和期望总报酬为

$$p^u(j|i) = \sum_a u(a|i)p(j|i, a),$$

$$\tau^u(i) = \sum_a u(a|i)\tau(i, a),$$

$$\text{POW}^u(i) = \sum_a u(a|i)\text{POW}(i, a).$$

假设动态电源管理过程是遍历的, 定义如下的平均能耗作为系统的优化指标:

$$J^u = \frac{\lim_{K \rightarrow \infty} \frac{1}{K} E^u \left\{ \sum_{n=0}^{K-1} \text{POW}(X_n, A_n) \right\}}{\pi^u \tau^u}, \quad (8)$$

其中  $\pi^u$  是转移矩阵  $P^u$  的稳态分布, 其满足方程组  $\pi^u P^u = \pi^u, \pi^u e = 1$ .

对于电源管理问题, 降低能耗不是唯一的目标. 如果只考虑降低能耗, 设备的工作性能将会下降, 甚至设备不能正常的工作. 为了保证设备的工作性能, 将唤醒时间视为延迟时间, 定义如下的平均延迟时间来衡量节能算法对设备性能的影响

$$C^u = \lim_{K \rightarrow \infty} \frac{1}{K} E^u \left\{ \sum_{n=0}^{K-1} c(X_n) \right\} = \frac{\pi^u c}{\pi^u \tau^u}, \quad (9)$$

其中  $c(X_n)$  表示在状态  $X_n$  时的延迟时间, 则有

$$c(\text{wake}) = T_{wu},$$

而在其他状态为0. 如果平均延迟时间较长, 表示系统较多时间用于了唤醒操作, 影响了设备的使用性能. 为了减少对设备性能的影响, 要求  $C^u$  小于给定的指标  $G$ . 因而, 动态电源管理问题可建模为如下的约束优化问题

$$\begin{aligned} \min J^u, \\ \text{s.t. } C^u \leq G. \end{aligned} \quad (10)$$

在模型参数已知的情况下, 线性规划可以用来求解上述问题的最优解<sup>[9]</sup>. 然而模型参数如转移概率, 平均逗留时间等需要繁琐的统计分析和计算. 如果转移概率和逗留时间计算不精确, 将会影响由此得到的最优电源管理策略. 此外, 考虑到用户对设备的使用习惯往往是不同的, 因而模型参数都是不同的, 对每个用户进行统计分析建模是不现实的事情, 为此本文提出了一种在线的优化技术.

### 3 策略参数化(Policy parametrization)

为实现管理策略的在线优化, 需要将管理策略  $u$  参数化, 用  $u(\alpha)$  来表示参数化后的策略, 其中  $\alpha$  是策略中的参数. 考虑到随机策略对于所有的  $i \in S$  需满足

$$\sum_{a \in A} u(a|i, \alpha) = 1. \quad (11)$$

在本文中, 主要对两种参数化方法进行比较分析. 第1种参数化为常用的soft-max参数化, 即:

$$u(a|i, \alpha) = \frac{\exp(\alpha_{i,a})}{\sum_{a \in A} \exp(\alpha_{i,a})}.$$

因而对于每一个空闲状态 $i$ , 需要一个二维的参数向量 $\alpha_i = (\alpha_{i,a_1}, \alpha_{i,a_2})$ . 在TISMDP模型中有 $M$ 个空闲状态, 该参数化方法需要 $2M$ 个参数. 考虑的第2种参数化方法是球面参数化方法<sup>[13]</sup>. 令 $u(a|i, \alpha) = \lambda_{i,a}^2$ , 由式(11), 有

$$\sum_{a \in A} \lambda_{i,a}^2 = 1,$$

这说明 $(\lambda_{i,a}, a \in A)$ 是位于单位球上的点. 所以通过球面坐标来参数化策略 $u$ , 即:

$$\begin{cases} u(a_1|i, \alpha_i) = \cos^2 \alpha_i, \\ u(a_2|i, \alpha_i) = \sin^2 \alpha_i. \end{cases} \quad (12)$$

该参数方法在每个状态 $i$ 处仅需要一个参数 $\alpha_i$ , 因而对于TISMDP模型需要 $M$ 个参数.

经过参数化之后, 相应的量可用参数形式描述. 如状态转移矩阵 $P(\alpha)$ , 期望逗留时间 $\tau(\alpha)$ , 平均功耗 $J(\alpha)$ 和平均延迟时间 $C(\alpha)$ . 因而, 优化问题(10)转变为下述关于参数 $\alpha$ 的约束优化问题

$$\begin{aligned} \min \quad & J(\alpha), \\ \text{s.t.} \quad & C(\alpha) \leq G. \end{aligned} \quad (13)$$

#### 4 在线优化算法(On-line optimization)

为了处理式(13)中的约束问题, 本文在文献[14]的基础上给出两种优化算法: 直接优化算法和增广拉格朗日乘法法, 并对它们进行比较分析.

##### 4.1 直接优化算法(Direct optimization)

该方法的主要思想是利用系统的样本轨迹在线估计平均功耗和平均延迟时间关于策略参数的梯度, 然后在约束范围内对平均能耗进行梯度搜索, 以便最小化平均能耗目标函数, 但如果参数更新到可行域外, 为使得(13)中的约束成立, 对平均延迟约束指标进行梯度搜索, 将参数拉回到可行域, 然后继续对平均功耗进行梯度寻优. 具体算法步骤如下:

**步骤1** 设置初始策略参数 $\alpha_0$ , 迭代次数 $N$ ,  $\beta \in (0, 1)$ 以及终止条件 $\varepsilon$ , 并令 $l = 0$ .

**步骤2** 设置 $n = 0$ , 并初始化 $Z_0 = 0, J_0 = 0, C_0 = 0, \Delta_0 = 0$ 以及 $\Delta_0^c = 0$ . 在策略 $u(\cdot|\alpha_l)$ 下, 得到系统的样本轨迹 $\{X_0, A_0, \tau_0, X_1, A_1, \tau_1, \dots, X_N, A_N, \tau_N\}$ , 重复如下迭代过程直到 $n = N$ , 得到平均能耗和平均延迟时间的估计值 $J_N$ 和 $C_N$ 以及它们的梯度的估计值 $\Delta_N$ 和 $\Delta_N^c$ .

$$Z_{n+1} = \beta Z_n + \frac{\nabla u(A_n|X_n)}{u(A_n|X_n)},$$

$$\begin{aligned} J_{n+1} &= J_n + \frac{1}{\tau_{n+1}} \{f(X_n, A_n) + \\ &\quad [P_{\text{ow}}(X_n) - J_n] (\tau_{n+1} - \tau_n)\}, \\ C_{n+1} &= C_n + \frac{1}{\tau_{n+1}} \{c(X_n, A_n) - C_n (\tau_{n+1} - \tau_n)\}, \\ \Delta_{n+1} &= \Delta_n + \gamma_{n+1} \{f(X_n, A_n) + \\ &\quad [P_{\text{ow}}(X_n) - J_{n+1}] (\tau_{n+1} - \tau_n) Z_{n+1} - \Delta_n\}, \\ \Delta_{n+1}^c &= \Delta_n^c + \gamma_{n+1} \{c(X_n, A_n) - \\ &\quad C_n (\tau_{n+1} - \tau_n) Z_{n+1} - \Delta_n^c\}, \end{aligned}$$

其中:  $\nabla u(A_n|X_n)$ 是策略 $u(A_n|X_n)$ 关于 $\alpha$ 的梯度,  $\gamma_n$ 是学习步长. 当 $n$ 趋向于无穷时,  $J_n$ 和 $C_n$ 分别给出了 $J(\alpha)$ 和 $C(\alpha)$ 的估计,  $\Delta_n$ 和 $\Delta_n^c$ 分别给出了 $J(\alpha)$ 和 $C(\alpha)$ 关于 $\alpha$ 的梯度估计.

**步骤3** 如果 $C_N \leq G$ , 利用 $\alpha_{l+1} = \alpha_l - \varepsilon_l \Delta_N(\text{mod } 2\pi)$ 更新 $\alpha_l$ , 否则, 利用 $\alpha_{l+1} = \alpha_l + \varepsilon_l \Delta_N^c(\text{mod } 2\pi)$ 更新 $\alpha_l$ , 其中 $\varepsilon_l$ 是步长, 采用球面参数化时, 对参数进行 $\text{mod } 2\pi$ 运算. 如果 $\|\alpha_{l+1} - \alpha_l\| \leq \varepsilon$ , 停止. 否则, 令 $l = l + 1$ , 然后转到第2步.

##### 4.2 增广拉格朗日乘法法(Augmented Lagrangian multiplier)

为了处理约束条件, 引入松弛变量 $z$ 把带不等式约束的优化问题(13)转化成带有等式约束的优化问题

$$\begin{aligned} \min \quad & J(\alpha), \\ \text{s.t.} \quad & C(\alpha) - G + z^2 = 0. \end{aligned}$$

定义

$$B(\psi) = C(\alpha) - G + z^2,$$

其中 $\psi$ 是 $\alpha$ 和 $z$ 的组合向量. 定义拉格朗日函数为

$$L(\psi) = J(\alpha) + \lambda B(\psi),$$

其中 $\lambda$ 为拉格朗日乘子. 为了保证 $L(\psi)$ 有局部凸性以便在线算法有较好的收敛性, 在拉格朗日函数上加入关于等式约束的惩罚项

$$L(\psi) = J(\alpha) + \lambda B(\psi) + \frac{\rho}{2} B^2(\psi), \quad (14)$$

其中 $\rho$ 是比较大的正常数. 对(14)式中的 $L(\psi)$ 关于 $\alpha$ 的求梯度得到

$$\frac{\partial L(\psi)}{\partial \alpha} = \frac{\partial J(\alpha)}{\partial \alpha} + \frac{\partial B(\psi)}{\partial \alpha} (\lambda + \rho B(\psi)). \quad (15)$$

由于 $\frac{\partial B(\psi)}{\partial \alpha} = \frac{\partial C(\alpha)}{\partial \alpha}$ , 故式(15)可以改写为

$$\frac{\partial L(\psi)}{\partial \alpha} = \frac{\partial J(\alpha)}{\partial \alpha} + \frac{\partial C(\alpha)}{\partial \alpha} (\lambda + \rho B(\psi)), \quad (16)$$

$L(\psi)$ 关于 $z$ 的梯度可以通过下式得到

$$\frac{\partial L(\psi)}{\partial z} = \frac{\partial J(\alpha)}{\partial z} + 2\lambda z + 2\rho z B(\psi). \quad (17)$$

由于  $\frac{\partial J(\alpha)}{\partial z} = 0$ , 故式(17)可以写为

$$\frac{\partial L(\psi)}{\partial z} = 2\lambda z + 2\rho z B(\psi). \quad (18)$$

基于增广拉格朗日乘数法的在线优化算法的主要思想是: 首先在线估计  $\frac{\partial J(\alpha)}{\partial \alpha}$  和  $\frac{\partial C(\alpha)}{\partial \alpha}$ , 然后根据(16)和(18)估计得到增广拉格朗日函数的梯度, 最后利用估计的梯度在线的更新策略参数  $\alpha$ . 详细的算法步骤如下:

**步骤 1** 设置初始策略参数  $\alpha_0$ , 松弛变量  $z_0$ , 拉格朗日乘子  $\lambda_0$ , 惩罚因子  $\rho$ , 迭代次数  $N$ , 以及终止条件  $\varepsilon$ , 并令  $l = 0$ .

**步骤 2** 与直接优化算法的第 2 步相同.

**步骤 3** 通过式  $B_N = C_N - G + z_l^2$  得到  $B(\psi)$  的估计值. 利用下面的迭代分别更新策略参数  $\alpha_l$ , 松弛变量  $z_l$  和拉格朗日乘子  $\lambda_l$ ,

$$\begin{aligned} \alpha_{l+1} &= [\alpha_l - \varepsilon_l (\Delta_N + \Delta_N^c (\lambda_l + \rho B_N))] \pmod{2\pi}, \\ z_{l+1} &= z_l - \gamma_l^z (2z_l (\lambda_l + \rho (C_N - G + z_l^2))), \\ \lambda_{l+1} &= \max [0, \lambda_l + \varepsilon_l B_N], \end{aligned}$$

其中  $\gamma_l^z$  和  $\varepsilon_l$  是步长. 如果  $\|\alpha_{l+1} - \alpha_l\| \leq \varepsilon$ , 停止. 否则, 令  $l = l + 1$ , 然后转到第 2 步.

从上述两种算法可以看出, 算法中的计算仅需要系统的运行轨迹, 不需要系统的精确模型, 因而避免了参数建模的复杂计算和统计分析以及参数不准确带来的影响. 此外, 算法不需要设备进行实时计算, 只需要对运行轨迹进行记录, 等到设备充电时利用记录的运行数据展开相关计算, 从而避免了计算所带来的额外能量消耗. 由于算法中的计算是都在设备上进行, 不需要将设备运行数据上传到云端, 因而避免了用户数据泄露的风险.

### 5 仿真结果(Results and analysis)

在这一小节中, 应用文献[6]中的硬盘电源管理问题来验证相应的算法, 设备参数如表 1 所示. 假设硬盘的服务率是 125, 当硬盘在空闲状态时的用户请求到达时间间隔服从帕累托分布(3), 分布参数为  $a = 0.9$ ,  $b = 0.65$ , 该硬盘系统的最小空闲时间为  $T_{be} = 5.43$  s, 此外, 该硬盘的最小有效休眠时间为  $T_{ms} = 3.15$  s.

表 1 硬盘参数

Table 1 The parameters of hard disk

$P_{ow}(\text{idle})$	$P_{ow}(\text{sleep})$	$T_{sd}$	$E_{sd}$	$T_{wu}$	$E_{wu}$
0.95 W	0.13 W	0.67 s	0.36 J	1.61 s	4.39 J

在算法中, 设置  $G = 0.1$ ,  $\beta = 0.8$  和  $N = 10000$ , 在增广拉格朗日乘数法中设置  $\rho = 50$ ,  $\lambda_0 = 1$  和  $z_0 = 1$ . 为了对比分析, 首先考虑不加入禁止时间的情形.

为了方便起见, 称直接优化算法为非增广拉格朗日函数方法. 根据不同的参数化方法, 考虑 4 种算法: 基于球面坐标参数化的增广拉格朗日函数算法(L-C)和非拉格朗日函数算法(N-L-C), 以及基于 soft-max 参数化的增广拉格朗日函数算法(L-S)和非拉格朗日函数算法(N-L-S). 图 2 中的仿真结果给出了 4 种算法随着迭代步数的平均能耗的变化情况.

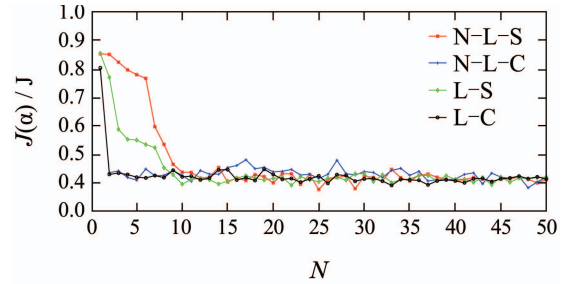


图 2 四种算法的平均功耗

Fig. 2 Power consumption of four algorithms

由图 2 可以看出, 整体来讲, 基于球面坐标的参数化方法比 soft-max 参数化方法有更快的收敛率. 此外, 相对于基于 soft-max 函数参数化的算法(N-L-S 和 L-S), 基于球面坐标参数化的算法(N-L-C 和 L-C)的参数数量少了一半, 因而算法第 2 步的计算时间可减少很多. 因此, 在后面的分析中, 主要考虑基于球面坐标参数化方法. 从图 2 可以看出, 增广拉格朗日乘数法(L-C)有更加平滑的收敛过程.

为了分析禁止时间的影响, 考虑禁止时间分别为 1 s, 2 s, ..., 10 s 等 10 种情况. 在此基础上, 利用 L-C 算法优化空闲状态的随机策略, 简记带有禁止时间的 L-C 算法为 T-L-C 算法. 首先仿真了 10 种情况下的 T-L-C 的能耗优化情况, 如图 3 所示, 从图中可以看出, 禁止时间设为 2 秒时具有最少的平均能耗. 图 4-6 分别给出了禁止时间为 2 s 时的 T-L-C 算法和 L-C 算法在算法迭代过程中, 平均能耗、平均延迟时间和错误休眠率的变化情况, 可以看出 T-L-C 算法得到了更低的能耗、更少的延迟时间和更低的错误休眠率.

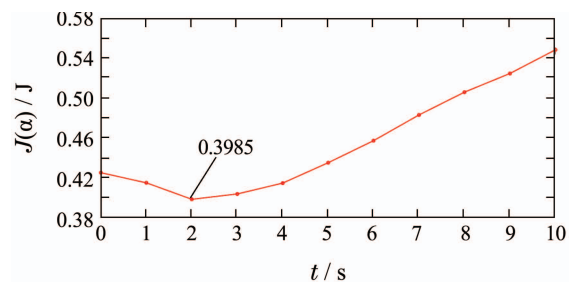


图 3 带有 10 种禁止时间的平均功耗

Fig. 3 The average power consumption with 10 forbidden times

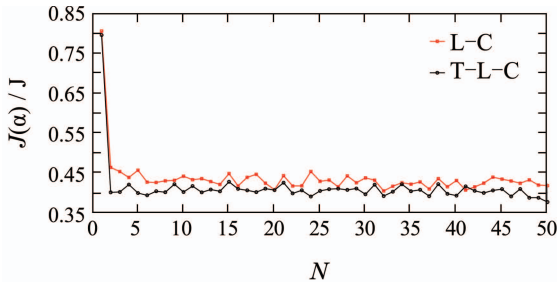


图4 T-L-C与L-C算法的平均能耗

Fig. 4 The average power consumption of T-L-C and L-C

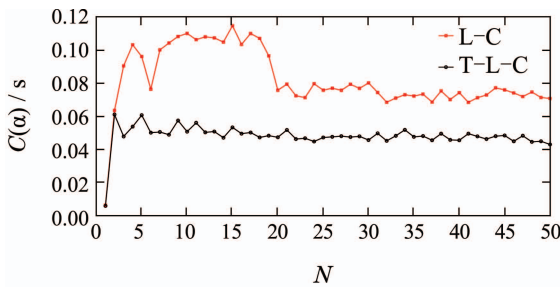


图5 T-L-C与L-C算法的平均延迟时间

Fig. 5 The average delay time of T-L-C and L-C

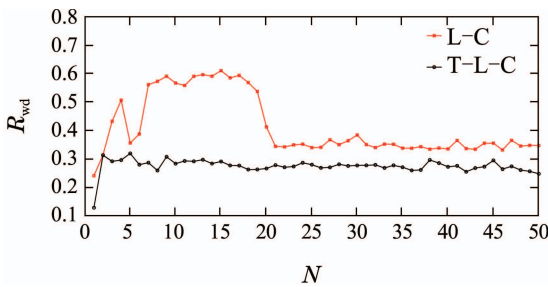


图6 T-L-C与L-C算法的错误休眠率

Fig. 6 The wrong sleep rate of T-L-C and L-C

## 6 结论(Conclusions)

在本文中,针对动态电源管理问题,结合两种不同的参数化方法和在线梯度估计方法提出了一种带有禁止时间的在线优化算法.该算法不依赖于模型参数并能够应用于动态电源管理的在线优化问题.该方法通过设备自身运行数据,自主地学习并改进电源的动态管理策略,从而可以使每台设备具有个性化的动态电源管理策略.由于优化过程可以集中在设备充电时完成,因而不需要通过云传输和计算,避免了隐私数据的泄露.增广拉格朗日乘数法处理约束条件使最优策略收敛过程更为平稳.通过对比两种不同参数化的方法,还可以发现参数化方法对算法的收敛性和计算时间有重要影响,球面参数化相比softmax参数化方法具有更好的收敛性,如何找到最好的参数化方法需要进一步的研究.此外,如何在线地优化选择禁止时间以及将方法推广到系统级节能管理等都是需要进一步研究的问题.

## 参考文献(References):

- [1] BENINI L, MICHELI G D. *Dynamic Power Management, Design Techniques and CAD Tools* [M]. Norwell, USA: Kluwer Academic Publishers, 1997.
- [2] LIU Fagui, MAI Weipeng, HUANG Kaiyao. Extension and realization of dynamic power management architecture [J]. *Journal of South China University of Technology (Natural Science Edition)*, 2007, 35(5): 60 – 64. (刘发贵, 麦伟鹏, 黄凯耀. 动态电源管理框架的扩展与实现 [J]. 华南理工大学学报(自然科学版), 2007, 35(5): 60 – 64.)
- [3] HWANG C H, WU A. A predictive system shutdown method for energy saving of event-driven computation [C] // *IEEE/ACM International Conference on Computer Aided Design*. San Jose, USA: IEEE, 1997: 28 – 32.
- [4] BENINI L, BOGLIOLO A, PALEOLOGO G A, et al. Policy optimization for dynamic power management [J]. *IEEE Transaction on Computer Aided Design Integrated Circuits and Systems*, 1999, 18(6): 813 – 833.
- [5] CHUNG E Y, BENINI L, PALEOLOGO G A, et al. Dynamic power management for non-stationary service requests [J]. *IEEE Transaction on Computer*, 2002, 51(11): 1345 – 1361.
- [6] OKAMURA H, MIYATA S, DOHI T. A Markov decision process approach to dynamic power management in a cluster system [J]. *IEEE Access*, 2016, 3: 3039 – 3047.
- [7] QIU Q, PEDRAM M. Dynamic power management based on continuous-time Markov decision processes [C] // *Proceedings of the 36th Design Automation Conference*. New Orleans, USA: IEEE, 1999: 555 – 561.
- [8] YIN Baoqun, LI Yanjie, TANG Hao, et al. Relations between discounted models and average models for semi-Markov decision processes [J]. *Control Theory & Applications*, 2006, 23(1): 65 – 68. (殷保群, 李衍杰, 唐昊, 等. 半Markov决策过程折扣模型与平均模型之间的关系 [J]. 控制理论与应用, 2006, 23(1): 65 – 68.)
- [9] SIMUNIC T, BENINI L, GLYNN P, et al. Event-driven power management [J]. *IEEE Transaction on Computer Aided Design of Integrated Circuits and Systems*, 2001, 20(7): 840 – 857.
- [10] JIANG Q, XI H S, YIN B Q. Adaptive optimisation of timeout policy for dynamic power management based on semi-Markov control processes [J]. *IET Control Theory & Applications*, 2010, 4(10): 1945 – 1958.
- [11] PUGHAT A, SHARMA V. A review on stochastic approach for dynamic power management in wireless sensor networks [J]. *Human-centric Computing and Information Sciences*, 2015, 5(1): 1 – 14.
- [12] ZHAI J F, LI Y J, CHEN H Y. An online optimization for dynamic power management [C] // *IEEE Conference on Industrial Technology*. Taipei: IEEE, 2016: 1533 – 1538.
- [13] ABAD F V, KRISHNAMURTHY V, MARTIN K, et al. Self-learning control of constrained Markov chains – a gradient approach [C] // *Proceedings of the 41st IEEE Conference on Decision and Control*. Las Vegas, USA: IEEE, 2002: 1940 – 1945.
- [14] LI Y J, CAO F. A basic formula for performance gradient estimation of semi-Markov decision processes [J]. *European Journal of Operational Research*, 2013, 224(2): 333 – 339.

## 作者简介:

翟剑锋 (1990–), 男, 硕士研究生, 目前研究方向为动态电源管理, E-mail: 641718702@qq.com;

李衍杰 (1978–), 男, 副教授, 目前研究方向为随机离散事件动态系统、智能决策与优化、强化学习, E-mail: autolyj@hit.edu.cn;

陈浩耀 (1981–), 男, 副教授, 目前研究方向为视觉感知与优化, E-mail: hychen5@hit.edu.cn.