

文章编号:1000-582X(2011)08-086-04

# 能耗限制的实时周期任务可靠性感知调度

罗 钧,刘永锋,付 丽

(重庆大学 光电技术及系统教育部重点实验室,重庆 400044)

**摘 要:**针对动态电压调节(Dynamic Voltage Scaling,DVS)技术在实时系统能耗管理中导致系统可靠性下降的问题,提出了基于人工蜂群算法(Artificial Bee Colony Algorithm,ABC)的静态/动态可靠性感知调度算法。该算法针对处理器频率离散可调的实时系统,利用人工蜂群算法原理在满足系统能耗约束和任务时限约束的同时尽可能地提高系统的可靠性。仿真结果表明,动态可靠性感知调度算法相比静态算法性能更优,接近负载预知的最优可靠性感知调度算法性能。

**关键词:**实时系统;能耗管理;可靠性;人工蜂群算法

**中图分类号:**TP316.7

**文献标志码:**A

## Reliability-aware schedule of periodic tasks in energy-constrained real-time systems

LUO Jun, LIU Yong-feng, FU Li

(Key Laboratory for Opto-electronic Technology & System, Ministry of Education,  
Chongqing University, Chongqing 400044, P. R. China)

**Abstract:** The Dynamic Voltage Scaling (DVS) technology which is widely used in numerous energy management schedule has negative effect on system reliability. Based on the artificial bee colony algorithm (ABC), two noble reliability-aware schedule algorithms are proposed for DVS systems with discrete frequencies, which will meet the energy constraint and deadline naturally while maximize the reliability of the system. The simulation results indicate that the dynamic schedule algorithm outperforms the static schedule algorithm, and its performance is close to that of the optimal scheduler that knows the exact workload in advance.

**Key words:** real-time system; energy management; reliability; artificial bee colony algorithm

随着电池供电嵌入式系统的广泛应用,不断提升系统性能所带来的能耗需求与有限的电池容量之间的矛盾日益突出。DVS是一种有效的节能技术,广泛应用于能耗管理中<sup>[1]</sup>,它通过降低处理器的电压和频率来降低系统能耗,但降低处理器频率会使任务的执行时间延长,甚至超过任务截止期,这在实

时系统中是无法接受的<sup>[2-3]</sup>。同时,降低处理器频率还会导致系统可靠性下降<sup>[4]</sup>。由此可见,在DVS能耗管理中,不仅要关注系统的实时性,还应充分考虑系统的可靠性。文献[4]研究了DVS对系统可靠性的影响,并建立了系统瞬时故障模型。在此基础上文献[5]提出了基于SUEF的启发式算法,文献[6]

**收稿日期:**2011-03-04

**基金项目:**重庆市经委科技攻关计划资助项目(10-cy-02);国防科工委国际军工计量“十一五”计划重点资助项目(J102006B104)

**作者简介:**罗钧(1963-),男,重庆大学教授,主要从事智能仪器系统、信息获取及处理研究与开发,  
(E-mail)luojun@cqu.edu.cn。

提出了 EAFT 算法,这些算法均是在保证系统实时性和可靠性的同时尽可能降低系统功耗。然而在军事、航天等领域,可靠性往往作为嵌入式系统最重要的指标优先考虑。因此,如何在满足系统能耗约束与任务时限约束的前提下最优化系统可靠性已成为当前面临的挑战。文献[7]考虑了理想化处理器模型,即处理器频率是连续可调的。而目前实际的处理器仅支持频率离散可调。笔者针对处理器频率离散可调的实时系统,在保证系统能耗约束和任务时序约束的同时最大化系统的可靠性。

## 1 系统模型及问题描述

### 1.1 系统功耗模型

文献[8]提出了嵌入式系统的功耗模型,执行单个任务  $T_i$  所需系统级能耗定义如下:

$$E_i(f) = P_{ind(i)} \frac{C_i}{f} + C_{ef} C_i f^2, \quad (1)$$

其中:  $C_i$  为任务  $T_i$  执行所需的时钟周期数;  $C_{ef}$  为系统相关常数;  $f$  表示执行  $T_i$  对应的处理器频率。  $P_{ind(i)}$  为  $T_i$  对应的频率无关的动态功耗。由式(1)可见,要降低系统能耗  $E_i(f)$ , 处理器频率  $f$  并不是越低越好,当处理器频率为  $f_{opt(i)}$  时,任务  $T_i$  的能耗最小。当处理器频率小于  $f_{opt(i)}$  时,系统能耗随频率的降低而急剧上升。

$$f_{opt(i)} = \sqrt[3]{\frac{P_{ind(i)}}{2C_{ef}}}. \quad (2)$$

### 1.2 瞬时故障模型

由于各种因素的影响(如硬件失效、电磁干扰、宇宙射线等),任务执行过程中系统可能产生瞬时故障导致任务失败<sup>[9-11]</sup>,使系统可靠性下降。系统瞬时故障率服从泊松分布,均值  $\lambda$  为处理器频率相关的函数<sup>[4]</sup>:

$$\lambda(f) = \lambda_0 10^{\frac{d(1-f)}{1-f_{\min}}}, \quad (3)$$

其中  $\lambda_0, d$  为系统相关的常数。  $f$  为处理器工作频率,且  $f \in [f_{\min}, f_{\max}]$ , 当  $f = f_{\min}$  时,  $\lambda_{\max} = \lambda_0 10^d$ 。由此可见,降低处理器频率会导致系统瞬时故障率均值呈指数上升。

### 1.3 任务模型

由  $N$  个实时周期任务组成的任务集为  $\Psi = \{T_1, T_2, \dots, T_N\}$ , 各个任务之间相互独立,在 0 时刻均处于就绪状态,任务  $T_i$  用四元组表示为  $\{C_i, P_{ind(i)}, P_i, R_i(f_i)\}$ 。

1) 在最坏情况下  $C_i = WCC_i$ , 其中  $WCC_i$  为  $T_i$  在最坏情况下执行所需的时钟周期数。

2)  $P_{ind(i)}$  表示任务  $T_i$  消耗的与频率无关的动态

功耗。

3)  $P_i$  为任务  $T_i$  的周期,设截止期  $d_i$  与  $P_i$  相等。

4)  $R_i(f_i)$  为单个任务  $T_i$  无故障执行的概率:

$$R_i(f_i) = e^{-\lambda(f_i) * \frac{C_i}{f_i}}. \quad (4)$$

由(4)式可知,任务  $T_i$  执行的处理器频率越高,  $R_i(f_i)$  越大。定义系统可靠性为所有任务无故障执行的概率:

$$R = \prod_{i=1}^n R_i(f_i). \quad (5)$$

所有任务均在一个可变电压处理器上执行,该处理器能在运行过程中动态地调整工作频率(忽略频率调整开销),假设处理器有  $M$  个离散的工作频率  $\{f_1, f_2, \dots, f_M\}$ , 其中  $f_{\min} = f_1, f_{\max} = f_M$  (归一化,令  $f_{\max} = 1$ )。

所有任务周期的最小公倍数即任务集的超周期记为  $T_{LCM}$ , 对于周期性任务集,每个超周期内的情况是相同的,所以只需在  $[0, T_{LCM}]$  范围内进行研究,每个任务在超周期内最多被调度一次。任务  $T_i$  在频率  $f_i$  下的执行时间为  $t_i(f_i)$ , 所需能耗为  $E_i(f_i)$ 。系统的总能耗约束记为  $E_{budget}$ 。

### 1.4 问题描述

系统的优化目标是为待调度任务集中的每个任务指派最优的处理器执行速度  $f_i$ , 达到系统可靠性最大化的目的:

$$\max \prod_{i=1}^n R_i(f_i). \quad (6)$$

并且所有任务执行的能耗满足系统能耗约束:

$$\sum_{i=1}^n E_i(f_i) \leq E_{budget}. \quad (7)$$

同时所有任务的执行时间不得超过其截止期:

$$d_i - \frac{C_i}{f_i} \geq 0, \quad (8)$$

其中,  $f_i \in [f_{opt(i)}, f_{\max}]$ 。由(3)(4)(5)(6)式,问题相当于在满足(7)(8)的条件下求解下式的最小值。

$$\min \sum_{i=1}^n \lambda(f_i) * \frac{C_i}{f_i}. \quad (9)$$

显然,以上描述相当于求解带约束条件的 NP 组合优化问题。

## 2 基于 ABC 的可靠性感知调度

### 2.1 ABC 算法模型

ABC 算法是建立在蜜蜂自组织模型和群体智能基础上的一种非数值优化计算方法,算法的目标是搜寻最优的蜜源,蜜蜂分 3 种角色:引领蜂(employed bees)、跟随蜂(onlookers)和侦察蜂

(scouts)。不同角色蜜蜂通过交流、转换和协作来实现群体智能<sup>[12]</sup>。

ABC算法在寻优等方面收敛速度快、非劣解质量高、鲁棒性好,非常适合求解复杂约束条件下组合优化问题<sup>[13]</sup>。

## 2.2 静态可靠性感知调度

在静态可靠性感知调度算法(Static Reliability-Aware Schedule, SRAS)中,假设任务  $T_i$  执行所需时钟周期数  $C_i = WCC_i$ 。

### 2.2.1 符号定义

1) 设蜂群规模(蜜源规模)为  $SN$ , 向量  $X_k = \{f_1, f_2, \dots, f_N\} (1 \leq k \leq SN)$  代表蜜源的位置, 即所求解。向量中  $N$  个元素分别对应待调度任务集中  $N$  个任务的执行频率。

2) 定义系统执行完所有任务的最小能耗  $E_{\min} = \sum_{i=1}^n E_i(f_{opt(i)})$ 。定义系统执行完所有任务的最大能耗  $E_{\max} = \sum_{i=1}^n E_i(f_{\max})$ 。系统能耗预算满足:  $E_{\min} \leq E_{\text{budget}} \leq E_{\max}$ 。

3) Modify算子: 由于ABC算法在追求系统可靠性最大化的同时必须满足能耗和时限约束, 因此, 为了避免迭代过程中产生不可行解, 每当蜜蜂发现新的蜜源时该算子立即检查该蜜源是否满足约束条件, 若不满足则将其修正为与当前状态最相近的可行解。

### 2.2.2 算法步骤

1) 设种群规模为  $SN$ , 由(10)式随机产生  $SN$  个蜜源的初始位置:

$$f_{k,i} = f_{opt(i)} + \text{rand} * (f_{\max} - f_{opt(i)}), \quad (10)$$

其中,  $f_{k,i}$  表示第  $k$  个可行解的第  $i$  个任务的执行频率。由此可得所有  $f_{k,i}$  组成的  $SN$  行  $N$  列的矩阵:

$$\begin{bmatrix} f_{1,1} & f_{1,2} & \cdots & f_{1,N} \\ f_{2,1} & f_{2,2} & \cdots & f_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ f_{SN,1} & f_{SN,2} & \cdots & f_{SN,N} \end{bmatrix}. \quad (11)$$

式中, 每一行代表一个蜜源的初始位置, 即所求问题的一个初始解。

2) 将引领蜂放到每个蜜源上, 利用(9)式计算每个蜜源的收益度  $fit$ 。

3) 引领蜂记住当前蜜源的收益度, 并在蜜源邻域内搜索新的蜜源, 用 Modify 算子修正。搜索公式如下:

$$X_{k,i} = X_{k,i} + (X_{k,i} - X_{j,i}) * r, \quad (12)$$

其中,  $k, j \in \{1, 2, \dots, SN\}$ , 且  $k \neq j$ ,  $r$  为  $[-1,$

$1]$ 之间的随机数。

4) 跟随蜂通过引领蜂获知蜜源收益度信息, 采用轮盘赌策略选择一个蜜源, 并根据式(12)在该蜜源邻域内搜索新的蜜源, 同样用 Modify 算子修正。选择概率  $P_k$  记为

$$P_k = \frac{fit_k}{\sum_{k=1}^{SN} fit_k}. \quad (13)$$

5) 在迭代过程中若新蜜源的收益度更优则更新原蜜源, 并记录迄今为止收益度最优的蜜源。

6) 若某一个引领蜂对应的蜜源在 Limit 次迭代后仍没有改进, 则该蜜源将被丢弃, 该引领蜂成为侦察蜂, 侦察蜂然后随机寻找新的蜜源, 并用 Modify 算子修正。

7) 判断终止条件是否满足。若满足则求出最优解, 若不满足转至步骤3)。

## 2.3 动态可靠性感知调度

SRAS算法给出了任务集在最坏情况下的静态调度方案。实际上并不是每个任务执行所需的实际时钟周期都为最坏情况下执行所需的时钟周期数, 若当前任务  $T_i$  的  $ACC_i$  小于  $WCC_i$ , 则表示任务  $T_i$  提前完成, 其实际消耗的能量必然小于最坏情况下所消耗的能量。由此产生的过剩能量可用来提高余下任务的执行速度, 从而进一步提高系统可靠性。

动态可靠性感知调度算法(Dynamic Reliability-Aware Schedule, DRAS)步骤如下:

1) 设当前等待执行的各个任务所对应的  $C_i = WCC_i$ , 执行 SRAS 算法为各任务分配初始执行速度。

2) 一旦有任务执行完毕, 则立即对比该任务的  $ACC_i$  和  $WCC_i$ , 并判断是否有过剩的能量可供回收。

3) 回收过剩能量, 并转至步骤1)直到所有任务执行完毕。

## 2.4 最优可靠性感知调度

负载预知的最优可靠性感知调度算法(Optimal Reliability-Aware Schedule, ORAS)预知任务执行过程中所需的实际时钟周期数。显然, 这在实际情况下是不可能实现的, ORAS 算法仅作其他算法的参考标准。

## 3 实验及性能分析

为了评估算法性能, 实验模拟了一个电池供电的嵌入式实时系统, 此系统集成了一个具备动态电压调节功能的 Intel Xscale 架构处理器, 该处理器有

5 个离散的归一化频率:  $\{0.15, 0.4, 0.6, 0.8, 1.0\}$ ,  $f_{\min} = 0.15, f_{\max} = 1.0$ 。可用 2 个相邻的频率来合成理论频率。WCC<sub>*i*</sub> 在  $[10, 100]$  之间随机产生。系统负载  $L = \sum_{i=1}^N \frac{ACC_i}{WCC_i}$ , 且  $L$  在  $[0.1, 1]$  之间变化。系统能耗约束  $\alpha = \frac{E_{\text{budget}}}{E_{\text{max}}}$  且  $\alpha \in [0.1, 1]$ 。P<sub>ind(i)</sub> 在  $[0.1, 0.8]$  之间随机生成, 且服从均匀分布。根据上述参数设置, 生成 100 个任务集, 每个任务集由 10 个任务组成, 以下各图中的每个点均为 100 次实验的平均值。

### 3.1 系统可靠性随能耗预算的变化情况

图 1 反映了系统负载为  $L=0.5$  时, 不同能耗约束下系统可靠性的变化情况。横坐标  $\alpha$  表示系统能耗约束, 纵坐标 PF 表示系统失效概率, PF 值越大则系统可靠性越低。

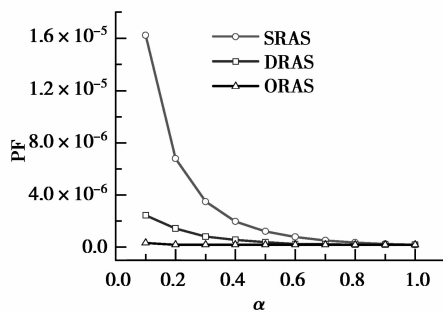


图 1 系统失效概率 VS. 系统能耗预算

由图 1 可见, 随着能耗预算的逐渐减小, 系统可靠性随之下降, 这是系统能耗预算的减少所必然导致的。当  $\alpha \in [0.8, 1]$  时, SRAS、DRAS、ORAS 算法性能保持一致。当  $\alpha \in [0.1, 0.8]$  时, DRAS 算法与 SRAS 算法相比更能保障系统可靠性, 其性能更接近 ORAS 算法。这与理论是相符的, 因为 DRAS 算法采用了能量回收机制, 进一步提高了系统的可靠性。

### 3.2 系统可靠性随系统负载的变化情况

图 2 反映了当  $\alpha=0.5$  时, 系统可靠性随系统负载的变化情况。横坐标  $L$  表示系统负载。

从图 2 可以看出, 随着系统负载的逐渐增大, 系统失效概率逐渐上升, 系统可靠性随之降低。这是因为系统负载越大, 任务执行的时间越长, 系统瞬时故障概率越高。由于 DRAS 算法采用了能量回收机制, 因此相对于 SRAS 算法而言, DRAS 性能更接近 ORAS 算法, 当负载  $L$  靠近 0.1 和 1 时, DRAS 算法性能曲线几乎与 ORAS 算法重合。值得注意的是当  $L=1$  时, 3 种算法所得系统可靠性完全一致, 因为此时所有任务的 ACC 均等于 WCC, 已没有

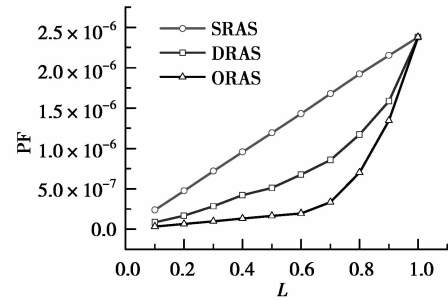


图 2 系统失效概率 VS. 系统负载

任务提前完成, 能量回收机制失效。

## 4 结 论

讨论了传统 DVS 能耗感知调度算法的不足, 针对处理器频率离散可调的实时系统提出了一种基于人工蜂群算法的可靠性感知调度算法。该算法为实时周期任务集中的每个任务分配处理器执行频率, 在满足任务实时性和系统能耗约束的同时尽量提高系统可靠性。理论分析及仿真结果表明相对于 SRAS 算法而言, 在相同条件下 DRAS 算法具有更优越的性能表现, 接近 ORAS 算法性能。

### 参考文献:

- [1] YAO F, DEMERS A, SHENKER S. Scheduling model for reduced CPU energy [C]//The 36<sup>th</sup> Annual Symposium on Foundations of Computer Science, October 23-25, 1995, Milwaukee, WI, USA. [USA]: IEEE, 2002:374-382.
- [2] RONG P, PEDRAM M. Energy-aware task scheduling and dynamic voltage scaling in a real-time system[J]. Journal of Low Power Electronics, 2008, 4(1):1-10.
- [3] AYDIN H, MELHEM R, MOSSE D, et al. Dynamic and aggressive scheduling techniques for power-aware real-time systems [C]//The 22<sup>nd</sup> IEEE Real-Time Systems Symposium, December 3-6, 2001, London, United Kingdom. [USA]:IEEE, 2002:95-105.
- [4] ZHU D, MELHEM R, MOSSE D. The effects of energy management on reliability in real-time embedded systems[C]//IEEE/ACM International Conference on Computer Aided Design, November 7-11, 2004, San Jose, CA, United States. [USA]:IEEE, 2005:35-40.
- [5] ZHU D, AYDIN H. Energy management for real-time embedded systems with reliability requirements [C]//IEEE/ACM International Conference on Computer-Aided Design, November 5-9, 2006, San Jose, CA, United States. [USA]:IEEE, 2007:528-534.

- 律的电法探测研究[J]. 岩石力学与工程学报, 2009, 28(2): 349-356.
- LIU SHU-CAI, LIU XIN-MING, JIANG ZHI-HAI, et al. Research on electrical prediction for evaluating water conducting fracture zones in coal seam floor[J]. Chinese Journal of Rock Mechanic and Engineering, 2009, 28(2): 349-356.
- [9] 王家臣, 许延春, 徐高明, 等. 矿井电剖面法探测工作面底板破坏深度的应用[J]. 煤炭科学技术, 2010, 38(1): 97-100.
- WANG JIA-CHEN, XU YAN-CHUN, XU GAO-MING, et al. Application of mine electric profiling method to detect floor failure depth of coal mining face [J]. Coal Science and Technology, 2010, 38(1): 97-100.
- [10] 赵贤任, 刘树才, 李富, 等. 煤层底板破坏带电阻率法异常特征研究 [J]. 工程地球物理学报, 2008, 5(2): 164-168.
- ZHAO XIAN-REN, LIU SHU-CAI, LI FU, et al. Study on resistivity abnormal characteristics of coal seam floor failure[J]. Chinese Journal of Engineering Geophysics, 2008, 5(2): 164-168.
- [11] 张平松, 刘盛东, 吴荣新, 等. 采煤面覆岩变形与破坏立体电法动态测试[J]. 岩石力学与工程学报, 2009, 28(9): 1871-1875.
- ZHANG PING-SONG, LIU SHENG-DONG, WU RONG-XIN, et al. Dynamic detection of overburden deformation and failure in mining workface by 3d resistivity method [J]. Chinese Journal of Rock Mechanic and Engineering, 2009, 28(9): 1871-1875.
- [12] 韩德品, 李丹, 程久龙, 等. 超前探测灾害性含导水地质构造的直流电法[J]. 煤炭学报, 2010, 35(4): 635-639.
- HAN DE-PIN, LI DAN, CHENG JIU-LONG, et al. DC method of advanced detecting disastrous water-conducting or water-bearing geological structures along same layer[J]. Journal of China Coal Society, 2010, 35(4): 635-639.
- (编辑 张小强)
- 
- (上接第 89 页)
- [6] ZHU P, YANG F, TU G, et al. Fault-tolerant scheduling for periodic tasks based on DVFS[C]//The 9th International Conference for Young Computer Scientists 2008, November 18-21, 2008, Zhangjiajie, Hunan, China. [USA]: IEEE, 2008: 2186-2191.
- [7] ZHAO B, AYDIN H, ZHU D. Reliability-aware dynamic voltage scaling for energy-constrained real-time embedded systems[C]// 26th IEEE International Conference on Computer Design 2008, October 12-15, 2008, Lake Tahoe, CA, United states. [USA]: IEEE, 2009: 633-639.
- [8] ZHU D, QI X, AYDIN H. Energy management for periodic real-time tasks with variable assurance requirements [C]//The 14th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications, 2008, August 25-27, 2008, Kaohsiung, Taiwan. [USA]: IEEE, 2008: 259-268.
- [9] CASTILLO X, MCCONNEL S R, SIEWIOREK D P. Derivation and calibration of a transient error reliability model[J]. IEEE Transactions on Computers, 1982, 31(7): 658-671.
- [10] IYER R K, ROSSETTI D J. A measurement-based model for workload dependence of cpu errors[J]. IEEE Transactions on Computers, 1986, 35(6): 511-519.
- [11] DABIRI F, AMINI N, ROFOUEI M, et al. Reliability-aware optimization for DVS-enabled real-time embedded systems [C]// The 9th International Symposium on Quality Electronic Design, March 17-19, 2008, San Jose, CA, United States. [USA]: IEEE, 2008: 780-783.
- [12] KARABOGA D, BASTURK B. Artificial Bee Colony (ABC) optimization algorithm for solving constrained optimization problems[C]//The 12<sup>th</sup> International Fuzzy Systems Association World Congress on Foundations of Fuzzy Logic and Soft Computing, June 18-21, 2007, Cancun, Mexico. [S. l.]: Springer, 2007: 789-798.
- [13] KARABOGA D, BASTURK B. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm [J]. Journal of Global Optimization, 2007, 39(3): 459-471.
- (编辑 陈移峰)