

细菌觅食优化算法在嵌入式系统多任务调度中的应用

吴 波,郝春梅

(哈尔滨金融学院 计算机系,黑龙江 哈尔滨 150030)

摘 要: 为了提高多核处理器任务调度的效率和负载均衡,提出一种改进细菌觅食优化算法的多核处理器任务调度方法.首先随机产生多核处理器任务有向无环图,然后构建一种多约束条件下的数学模型,最后采用改进细菌觅食优化算法对模型进行求解.仿真结果表明,改进细菌觅食优化算法很好避免了其它群智算法存在的缺陷,可以较好地获得全局最优调度方案,使执行任务的时间最短,很好地实现了多核处理器任务进行的分配和调度.

关键词: 多处理器;任务调度;负载均衡;细菌觅食优化算法

中图分类号: TP31

文献标识码: A

文章编号: 1000-7180(2013)03-0143-05

Task Scheduling of Multi-core Processor System Based on Improved Bacteria Foraging Optimization Algorithm

WU Bo, HAO Chun-mei

(Department of Computer Science, Harbin Finance University, Harbin 150030, China)

Abstract: In order to improve the efficiency and load balance of chip multiprocessor task scheduling, this paper proposed a task scheduling method based on improved bacteria foraging optimization algorithm. Firstly, task directed acyclic graph are randomly generated, and then a constrained mathematical model is constructed, finally improved bacteria foraging optimization algorithm is used to solve the mathematical model. The simulation results show that the proposed method can sole other defects, and can obtain the better scheduling scheme, and the task execution time is the shortest in all methods, and it can effectively achieve multiprocessor task allocation and scheduling.

Key words: multi-core processor; task scheduling; load balancing; bacteria foraging optimization algorithm

1 引言

多核处理器系统因其具有高效率、低成本等特点,广受人们的喜爱,并成为嵌入式系统中最常用的处理器系统.在对多核处理器系统研究的过程中,系统任务调度问题显得尤为重要,一个好的任务调度算法可以充分发挥系统性能^[1].

大量研究表明,多核处理器系统任务调度问题已经被证明是 NP 难题,采用传统穷举法或启发式算法进行求解,计算复杂度大,耗时长,有时甚至无法实现^[2].80 年代以来,学者们受到自然界规律启发,提出遗传算法、粒子群算法、蚁群算法等群智能

算法,这些算法搜索能力强、收敛速度快,在 NP 难题求解中得到广泛的应用^[3-4].苗蕾等提出了基于遗传算法的多核处理器任务调度策略,结果表明,遗传算法可以较好的实现多核处理器任务分配和调度^[6].杨洪斌提出了基于粒子群算法的多核处理器任务调度方法,并取得了不错的效果.然而在实际应用中,遗传算法和粒子群算法等群智能算法均具有各自的缺陷,如易陷入局部极值,这样就限制了它们在多核处理器任务高度中的应用^[6-7].2002 年 Passino 教授根据细菌的觅食和趋化行为,提出了用于解决优化问题的细菌觅食优化(Bacterial Foraging Optimization, BFO)算法,为群智能算法家族又添了

收稿日期:2012-06-20;修回日期:2012-07-20

基金项目:黑龙江省教育厅科学技术研究项目(1252101064)

新成员^[8].

为了获得更优的多核处理器系统任务分配和调度方案,利用细菌觅食优化算法的优点,文中提出一种基于改进细菌觅食优化算法(Improved Bacterial Foraging Optimization, IBFO)的多核处理器系统任务调度方法,并通过仿真实验进行验证.

2 多核处理器任务调度的数学模型

在多核处理器系统中,本研究认为各个计算节点是同构的,借鉴相关文献研究,多核处理器系统任务调度有如下定义:

定义 1 有一个任务被分割成 m 个子任务 $T = \{T_1, T_2, \dots, T_m\}$, 且任务之间有一约束条件, 该约束条件可以用一个有向无环图来描述, 具体如图 1 所示^[9].

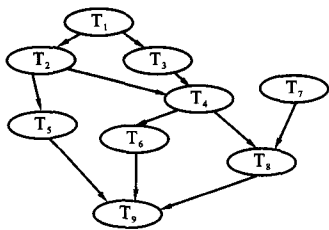


图 1 DAG 任务图

定义 2 设一个多核处理器系统包括 $n \times p$ 个同构的处理器核 $P = \{P_{ij} \mid i \in [1, n], j \in [1, p]\}$, 同一任务在任意处理器核上的执行时间相等, P_x 与 P_y 间的距离定义为 $\sqrt{(z-s)^2 + (x-t)^2}$.

定义 3 采用一维向量 $ET = \{ET_1, ET_2, \dots, ET_m\}$ 表示执行子任务集合 T 的预计运行时间使用, 其中 ET_j 表示子任务 T_j 所需要的时间.

定义 4 通信延迟与处理器核、任务 T_i 和 T_j 间传输延迟与传输数据量 $Comu_{ij}$ 均呈正比例关系, 所有子任务之间的传输数据量构成二维数组 $Comu_{m \times n}$.

定义 5 二维数组 $Sche = \{\sigma_{ijk} \in \{1, 0\} \mid i \in [1, n], j \in [1, p], k \in [1, m]\}$, $\sigma_{ijk} = 1$ 表示任务 T_k 在处理器核 P_{ij} 上执行, 集合 $Pre[T_i]$ 可以表示任务 T_i 的所有前驱任务.

定义 6 任务 T_j 的开始时间 $Begin[j]$, 完成时间 $Finish[j]$, 则可以得出:

$$Finish[k] = ET_k + Begin[k] \quad (1)$$

$$Begin[k] = \max_{T_p \in Pre[T_k]} \{Finish[p] +$$

$$\frac{\lambda}{\Delta} \sqrt{(s-i)^2 + (t-j)^2} (1 - \sigma_{ijk}) Comu_{pk} \}$$

式中, λ 是一个常数; T_p 为任务 T_k 的前驱任务;

$$\Delta = \sqrt{(n-1)^2 + (p-1)^2}.$$

多核处理器系统的任务调度目标是从策略集合 ST 中找到调度策略 St' , 使多核处理器系统执行任务 T 的时间最短, 因此, 多核处理器任务调度的数学模型可以描述为^[10]:

$$\begin{cases} \min \{ \max \{ Finish[T_i] \} \} \\ T_i \in T \\ ST' \in ST \end{cases} \quad (3)$$

BFO 算法通过群体中各个细菌间的合作与竞争而产生的群体智能指导优化搜索, 算法本身具备分布并行的寻优能力、对初值不敏感、较好的全局搜索能力等特点, 因此本研究采用 BFO 算法对式 (3) 进行求解, 并对其进行改进.

3 改进细菌觅食优化算法

3.1 基本细菌觅食优化算法

基本细菌觅食优化算法(BFO)模拟大量细菌在觅食过程中的一系列行为, 通过趋向性、复制、迁徙及群体感应等 4 个基本操作对问题进行求解, 当达到最大迭代次数或满足预设精度时算法结束.

3.1.1 趋向性操作

细菌在觅食过程中, 个体首先通过旋转转移到新的位置, 如果新的位置具有较好的适应值, 那么就沿着前次旋转的方向泳动, 直到适应值下降或达到最大泳动步数, 该过程称为趋向性操作.

设一个细菌种群规模为 S , 问题的一个候选解可以采用细菌所处的位置进行描述, 那么 $\theta = [\theta_1, \theta_2, \dots, \theta_D]$, $i = 1, 2, \dots$ 表示细菌 i 的 D 维向量, 其中 $\theta(j, k, l)$ 表示细菌 i 经过 k 次复制操作和 l 次迁徙操作之后的位置. 细菌 i 的每一步趋向性操作为:

$$\theta(j+1, k, l) = \theta(j, k, l) + C(i)\Phi(j) \quad (4)$$

其中, $\Phi(j)$ 表示一个随机前进方向, $C(i)$ 表示向前游动的步长单位.

3.1.2 复制操作

当所有的细菌完成一轮趋向性操作后, 将细菌个体依照在迭代过程中适应度值之和的优劣排序, 把排在后面的一半细菌淘汰掉, 设淘汰掉的细菌个数为 $S_r = S/2$, 剩余的 S_r 细菌自我复制生成与一个与自己具有相同的位置的新个体, 这个过程称为复制操作.

3.1.3 迁徙操作

迁徙操作有利于细菌个体跳出局部极值, 寻

找全局最优解.在算法执行过程中,细菌个体会依一定的概率 P_{ad} 突然消失,类似于细菌遭遇偶然事件死亡,在个体消失的同时,在解空间中任意产生一个新的个体,以保持个体总量的不变,这就是迁徙.

3.1.4 群体感应机制

个体在移动过程中还会释放一定的化学物质,向同伴传达自身所处位置的营养物浓度,以便于对方决定是否向该位置靠拢;同时附近个体发出的排斥力信号,因此每一个细菌个体受到自身的信息和其他个体的信息影响,即种群中其他细菌传递的觅食信息.这种现象称为感应机制.

设 $P(j, k, l) = \{\theta_i(j, k, l) \mid i = 1, 2, \dots, S\}$ 为细菌位置, $J(i, j, k, l)$ 表示适应度函数值,那么细菌之间传递信号的影响值是:

$$J_{\alpha}(\theta, p(j, k, l)) \sum_{l=1}^S J_{\alpha}(\theta, \theta'(j, k, l)) = \sum_{l=1}^S \left[-d_{\text{attractant}} \exp(-w_{\text{attractant}} \sum_{m=1}^D (\theta_m - \theta'_m)^2) \right] + \sum_{l=1}^S \left[h_{\text{repellant}} \exp(-w_{\text{repellant}} \sum_{m=1}^D (\theta_m - \theta'_m)^2) \right] \quad (5)$$

综合上述可知,细菌 i 执行一次趋向性操作后的新适应度值为:

$$J(i, j+1, k, l) = J(i, j, k, l) + J_{\alpha}(\theta'(j+1, k, l), P(j+1, k, l)) \quad (6)$$

其中, $w_{\text{repellant}}$ 表示斥力的宽度; $h_{\text{repellant}}$ 表示斥力的高度; $d_{\text{attractant}}$ 为引力的深度^[11-12].

3.2 BFO 算法的改进

3.2.1 迁徙操作的改进

在 BFO 算法中,通过迁徙,可以帮助陷入局部极值的细菌个体重新选择位置,从而跳出局部极值.由于迁徙是以一定概率随机发生的,因此迁徙操作同样有可能选中已经找到最优位置的细菌,使其发生迁徙而离开最优位置.对于已经找到最优值的细菌个体离开最优位置的现象,我们称之为“逃逸(escape)”.迁徙操作是能帮助算法在寻优过程中跳出局部极值;而迁徙操作又不可避免的会产生逃逸现象,这是我们所不愿看到的.为此,对迁徙操作进行改进.具体改进如下:

首先将细菌个体按照当前的适应度值进行排序,迁徙操作仅从适应度值排在后面的部分细菌个体中,按照概率 P_{ad} 进行选择,而排在前面的个体即为已经找到全局最优解或者接近全局最优解的个体,这些细菌个体完全不迁徙,这样就能有效的避免逃逸的发生.考虑到随着代数的增大,细菌个体不断

向最优位置靠拢,可发生随机迁徙细菌的比例应适当减小.设 Q 为可选择迁徙的细菌百分比,初值为 1, generation 为菌群的代数,初值为 0,在算法中设定每隔 10 代迁徙一次,隔 20 代缩小一次迁徙范围,即若 $\text{generation} \bmod 10 = 0$,则执行迁徙操作,令 $\text{ged} = \text{generation} \div 20$,此时 $Q = 1 - (2^{\text{ged}} * L)$, L 为不参与迁徙的细菌个体所占的百分比的初值,在按适应度值优劣排序后的细菌队列中,从后面 $(1 - (2^{\text{ged}} * L))\%$ 个细菌个体中按照概率 P_{ad} 进行迁徙,随着 generation 的增大,ged 增大,可选择迁徙的细菌比例减小,这样就可以避免逃逸的发生,极大的提高收敛速度.

3.2.2 复制操作的改进

在 BFO 算法中,通过复制操作,将适应值差的一半细菌淘汰,适应值好的一半细菌自我复制,各自生成与自身相同的一个新个体,从而保持细菌种群规模不变.复制操作以适应值高的细菌个体取代适应值低的,可以极大的提高算法的收敛速度.在 BFO 算法中,细菌均按照所有位置适应度值总和进行排序,实际上细菌的适应度值只与当前的位置有关,与细菌个体所经历的历史适应度值无关,因此,对复制操作进行改进,将细菌个体的排序以当前适应度值来进行,用细菌个体当前的最优位置来评价个体的优劣,进行复制操作,可以更有效的完成优胜劣汰,从而提高算法收敛速度.

3.3 IBFO 的工作流程

(1) 初始化菌群规模 S 、每个细菌的初始位置,同一方向最大泳动步数 try_number ,步长 step ,细菌个体间引力深度、斥力高度、引力宽度、斥力宽度,迁徙概率 P_{ad} ,初始不迁徙细菌百分比 L 等参数.

(2) 计算每个细菌的初始适应度值.

(3) 令 $i = 0, \text{generation} = 0, t = 0, Q = 1$.

(4) 趋向性操作与群体感应机制:

① 令 J_{last} 为当前细菌的适应度值;

② 选择一个随机方向泳动,得到新位置的适应度值 J_{next} ;

③ 按式(2) 计算其他细菌对本细菌的影响值 J_{cc} ,令 $J_{\text{next}} = J_{\text{next}} + J_{\text{cc}}$;

④ 若 $t < \text{try_number}$, $t = t + 1$,否则 $t = 0$,转⑦;

⑤ 若 J_{next} 优于 J_{last} , $J_{\text{last}} = J_{\text{next}}$,否则 $t = 0$,转⑦;

⑥ 在该方向上再走一步, $J_{\text{next}} =$ 新位置的适应度值,转④;

⑦ $i = i + 1$,若 $i < S$,转①;否则 $i = 0$, $\text{generation} = \text{generation} + 1$,转(5).

(5) 复制操作:

if generation MOD 5 $\neq$ 0, 转(7);

② 按各细菌当前的适应度值, 从优到劣排序;

③ 将适应值高的一半细菌的位置, 依次赋值给适应度低的一半细菌, 更新细菌的适应度值.

(6) 迁徙操作:

① if generation MOD 10 $\neq$ 0, 转(7);

② 令 $ged = \text{generation DIV } 20$, 计算当前可选择的迁徙范围 $Q = 1 - (2^{ged} * L)$, 若 $Q < 0$, 则令 $Q = 0$;

③ 按各细菌当前的适应度值, 从优到劣排序;

④ 令 $S_{ad} = S * Q$, 选择排序后细菌队列中后 S_{ad} 个细菌按照概率 P_{ad} 迁徙;

(7) 检查终止条件, 若满足终止条件, 则输出最优解, 算法终止; 否则转(4).

4 IBFO 在多处理器任务调度的应用

4.1 仿真环境及参数设置

仿真平台: CPU PIV/2.8 GHz, 内存 2GB, Windows XP, Matlab 2007, 采用 Simics 模拟软件在实现算法. 为了评价 IBFO 的多核处理器任务调度性能, 选择粒子群算法(PSO)和遗传算法(GA)作为对比算法. IBFO 算法参数如下: 种群规模为 30, 复制次数 $N_r=5$, 趋化次数 $N_c=10$, 游动步长 $C=0.01$; 迁徙次数 $N_{ad}=6$. PSO 和 GA 种群规模与 IBFO 相同, 最大迭代次数均为 500. PSO 参数为: $c_1=c_2=2.0, w=1$. GA 参数为: 交叉概率 0.7, 变异概率 0.2. 全部多核处理器任务调度算法均独立运行 10 次, 采用平均值作为最终结果.

4.2 结果与分析

4.2.1 不同任务规模调度性能比较

在多核处理器核数为 20 的情况下, 采用 IBFO、PSO、GA 对不同任务规模进行仿真测试, 得到的任务调度完成时间如图 2 所示.

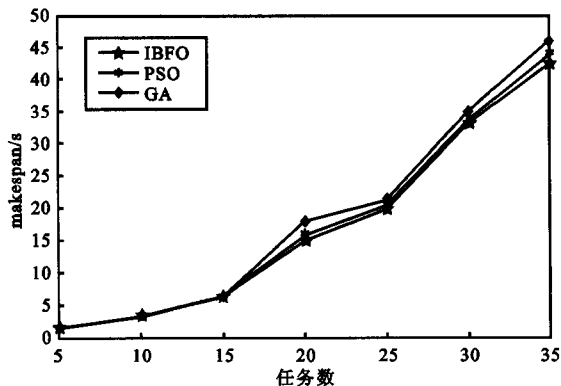


图 2 几种算法的任务调度性能对比

从图 2 可知, 当任务数不大时, 各种算法的调度性能没有明显差异. 随着任务数量增加, 任务调度完成时间均有开始增加, 其中 IBFO 增长曲线相对平缓, 增长幅度较小, 明显小于参比算法, 实验结果表明, 相对于参比算法, IBFO 加快了多核处理器任务调度方案求解速度, 可以找到全局最优多核处理器任务调度方案.

4.2.2 处理器负载均衡对比

各算法的多核处理器系统的负载如图 3 所示.

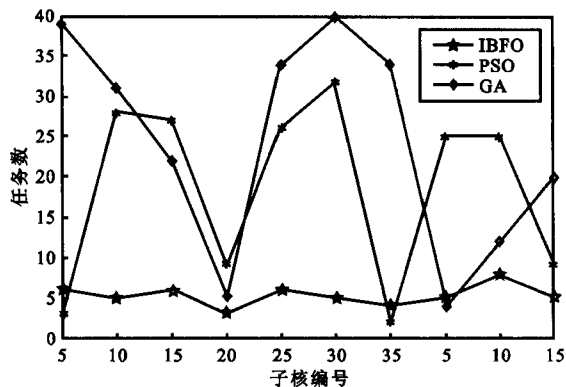


图 3 几种算法的负载均衡性能比较

从图 3 可知, 在所有任务调度算法中, 基于 IBFO 的多核处理器系统的负载最均衡, 而 PSO、GA 中系统中负载不均衡, 对比结果表明, IBFO 很好的保证多核处理器的负载平衡, 使多核处理器系统的资源利用率得以提高.

4.2.3 几种算法收敛性能比较

几种算法的最大收敛代数如表 1 所示. 由表 1 可知, 在所有多任务调度种算法中, IBFO 算法由于迁徙操作和复制操作进行改进, 收敛速度明显加快, 很好的保护了种群的多样性, 具有较好的寻优能力, 有效避免了对比算法的局部极值、后期收敛速度慢等缺陷.

表 1 几种算法的收敛速度比较

算法	收敛最优解代数
PSO	200
GA	250
IBFO	100

5 结束语

针对多核处理器任务调度的特点和当前调度算法存在的缺陷, 将新型的群智能算法—BFO 引入到多核处理器任务调度求解过程中, 并通过具体应用实例验证了算法的可行性和有效性. 但是 BFO 算法还不到十年, 还属于初始研究步阶段, 算法的参数值

是影响算法性能和效率的关键,因此如何确定BFOA的最优参数使多核处理器任务调性能更优是将来研究的方向。

参考文献:

- [1] Burger D, Goodman J R. Billion—Transistor architectures: there and back again [J]. IEEE Computer, 2004, 37, (3): 22-28.
- [2] Jo Ann M Paul, Donald E Thomas, Andrew S. Cassidy: high-level modeling and simulation of single-chip programmable heterogeneous multiprocessors [J]. ACM Transaction of Design Automation Electric System. 2005, 10(3): 431-46.
- [3] David Geer. Industry trends: chip makers turn to multi-core processors Computer, 2005, 38, (5): 11-13.
- [4] 苗蕾, 齐勇, 侯迪, 等. 基于遗传算法的片上多处理器任务调度策略研究[J]. 微电子学与计算, 2007, 24 (6): 8-11.
- [5] 杨洪斌, 陈伟, 吴悦. 基于粒子群算法的多核处理器线程调度研究[J]. 计算机工程与设计, 2010, 31 (5): 1045-1047.
- [6] 赵国材, 刘洋. 基于 MMAS 的多目标优化算法研究[J]. 计算机仿真, 2011, 28(7): 232-235.
- [7] 蒋建春, 汪同庆. 多核处理器的任务调度算法[J]. 计算机工程与应用. 2009, 45(33): 52-56.
- [8] Ravindran K. Task allocation and scheduling of concurrent applications to multiprocessor systems [D]. Berkeley: University of California, Berkeley, 2007.
- [9] 邱卫东, 陈燕, 李洁萍, 等. 一种实时嵌入式系统的任务调度算法[J]. 软件学报, 2004, 15(4): 504-511.
- [10] 徐成, 王立东. 基于遗传算法的多核多帧任务分配[J]. 计算机应用研究, 2009(11): 4066-4068.
- [11] 易军, 李太福. 求解作业车间调度的变邻域细菌觅食优化算法[J]. 机械工程学报, 2012, 38(12): 179-204.
- [12] 周雅兰. 细菌觅食优化算法的研究与应用[J]. 计算机工程与应用, 2010, 46(20): 16-19.

作者简介:

吴波 男, (1972—), 硕士, 副教授. 研究方向为计算机应用.

郝春梅 女, (1975—), 硕士, 副教授. 研究方向为计算机应用.

(上接第 142 页)

- [2] 赵铁. 往复式压缩机气阀故障诊断的实验研究[D]. 西安: 西北工业大学, 2003.
- [3] Isermann R. (1993). Fault diagnosis of machines via parameter estimation and knowledge processing: tutorial paper[J]. Automatica, 29(4): 815-835.
- [4] 徐章遂, 房立青, 王希武, 等. 故障信息诊断原理及应用[M]. 北京: 国防工业出版社, 2000.
- [5] 刘敏华. 基于 SDG 模型的故障诊断及应用研究[D]. 北京: 清华大学, 2005.
- [6] 杨帆, 萧德云. 基于 SDG 的复杂系统故障传播规律分析[J]. 高技术通讯, 2005, 15(10): 33. 36.
- [7] 徐新海. 基于层次 SDG 的航空发动机故障诊断方法研究[D]. 南京: 南京航空航天大学, 2010.

作者简介:

聂银燕 女, (1986—), 硕士研究生. 研究方向为控制工程.

林晓焕 女, (1964—), 教授, 研究生导师. 研究方向为数据通信与计算机控制.