

基于多核 PC 的人工蜂群并行算法的研究与实现

李 双, 李文敬, 杨 文, 周海燕

(广西师范学院 计算机与信息工程学院, 广西 南宁 530023)

摘 要: 针对人工蜂群算法搜索时间长、收敛速度缓慢、搜索后期极易陷入局部最优等问题, 提出了一种人工蜂群并行算法 (PRP-ABC)。首先引入随机扰动因子和全局最优解两个策略得到串行式 RP-ABC 算法; 然后应用 OpenMP 并行技术, 采用主从式调度方式对串行算法并行改写; 最后在多核 PC 上将三种算法在复杂函数优化问题上测试比较。结果表明, 该算法有效提高了收敛速度和优化性能, 尤其适用于高维复杂函数的优化问题。

关键词: 人工蜂群算法; 并行算法; 随机扰动因子; 多核

中图分类号: TP319

文献标识码: A

文章编号: 1000-7180(2012)09-0063-04

Research and Implementation of Parallel Random Perturbation Artificial Bee Colony Algorithm Based on Multi-core PC

LI Shuang, LI Wen-jing, YANG Wen, ZHOU Hai-yan

(School of Computer and Information Engineering, Guangxi Teachers Education University, Nanning 530023, China)

Abstract: As considering the problem of search time long, and easily into local optimal solution in later stage of evolution in Artificial Bee Colony (ABC) algorithm, a Parallel Random Perturbation Artificial Bee Colony (PRP-ABC) algorithm is proposed out. Firstly, introduced two concepts here which Random Perturbation factor and global optimal solution, get a serial type RP-ABC algorithm; Secondly, using OpenMP parallel rewrite it, this paper using master-slave scheduling model; Finally, tested the three algorithm about complex function optimization problem under the multi-core PC environment. The result shows that the algorithm convergence speed and the optimization performance had been improved, especially suitable for the high dimension of complex function optimization problems.

Key words: artificial bee colony; parallel algorithm; random perturbation; multi-core

1 引言

Karaboga 于 2005 年提出了基于 Seeley 的蜂群自组织模型的人工蜂群算法 (Artificial Bee Colony algorithm, ABC)^[1], 该算法是一种模拟蜜蜂采蜜行为的群体智能优化算法, 通过蜂群的分工实现目标搜索, 具有参数设置简单、易于实现等特点。文献[2-3]对 ABC 算法进行了改进, 在一定程度上提高了算法的搜索性能, 但是现有的 ABC 算法很多都是串行的, 对于大规模复杂工程优化问题就需要发挥蜂群算法本身的并行性, 多核 PC 的普及为人工蜂群算

法的并行化提供了实现途径。

2 对人工蜂群算法的改进

2.1 人工蜂群算法基本思想

人工蜂群算法 (ABC) 由引领蜂 (employ bees), 观望蜂 (onlooker bees) 和侦查蜂 (scout bees) 三部分组成, 解的群体由 SN 个 D 维向量表示, 第 i 个解可以表示为 x_i , $x_i = (x_{i1}, x_{i2}, \dots, x_{iD})$, $i = 1, 2, \dots, SN$, 食物源花粉量对应解的质量 (适应度值)。搜索流程如下:

(1) 引领蜂根据公式 (1) 生成一个新食物源进

收稿日期: 2012-02-12; 修回日期: 2012-03-08

基金项目: 国家自然科学基金项目 (61163012); 广西自然科学基金项目 (0991105)

行比较选取:

$$v_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj}) \quad (1)$$

其中, v_{ij} 是新位置, k 随机选取, $k \in \{1, 2, \dots, BN\}$ (BN 是种群个数), 且 $k \neq i, j \in \{1, 2, \dots, D\}$ (D 为目标空间的维数), ϕ 是 $[-1, 1]$ 之间的随机数^[4].

(2) 计算引领蜂所搜索食物源质量的概率值 P_i , 观望蜂根据公式(2)选择一个食物源 $x(i)$:

$$P_i = \text{fit} / \sum_{n=1}^{SN} \text{fit}_n \quad (2)$$

其中, fit 为第 i 只引领蜂所对应食物源的适应度值.

(3) 观望蜂根据公式(1)产生新解 $v(i)$;

(4) 食物源未更新次数达到限定次数 limit 后, 引领蜂放弃食物源重新搜索; 更新公式:

$$x_i = \min_j + \text{rand}(0, 1)(\max_j - \min_j) \quad (3)$$

2.2 随机摄动因子策略

根据文献[5]的思想在式(2)中引入随机摄动因子, 提出了 RP-ABC 算法 (Random Perturbation-ABC, RP-ABC), 改进过程如下:

(1) 引入全局最优解 gbest 参数, 设立最优解公告牌, 仅对观望蜂可见;

(2) 记录当前全局最优解位置;

(3) 引入随机摄动因子 η , 食物源更新公式:

$$v_{ij} = x_{ij} + \eta \phi_{ij} \quad (4)$$

其中, $\eta \phi_{ij}$ 项使蜜蜂产生一个围绕 x_{ij} 的随机波动, 当前位置 x_{ij} 与最优位置 gbest_{ij} 的间距以及蜂群规模决定 η 的取值:

$$\eta = \begin{cases} (u_b - l_b)/N & \text{if } \epsilon > (f_{\text{pbest}} - f_{\text{pbest}+t})/f_{\text{pbest}} \\ (\text{gbest}_{ij} - x_{ij})/N & \text{else} \end{cases} \quad (5)$$

其中, (u_b, l_b) 为搜索空间; N 为常数, 当波动范围一定时, N 随种群规模成正比变化, 使之更快的靠近最优食物源位置; ϵ 为预先设定的常数阈值; f_{pbest} 为目标函数的局部最优值; $f_{\text{pbest}+t}$ 为经过 t 次循环后目标函数的 f_{pbest} 值. 当目标函数循环 limit 次后, 其相对变化率仍小于 ϵ , 则重新初始化该处食物源, 使其摄动半径遍布定义区间 $[u_b, l_b]$.

3 并行算法的分析与设计

3.1 串行 RP-ABC 算法性能分析

从以上描述可以预计在高维空间下引领蜂与观望蜂对邻域循环搜索选择耗时最多, 使用 Intel Parallel Studio 的 Parallel Amplifier 组件对串行 RP-ABC 算法在 50 维下进行性能分析, 找出耗时多的

程序段进行下一步的并行改写. 结果如表 1.

表 1 对串行 ABC 算法的性能分析表

Function	CPU Time/s
SendOnlookerBees()	8.280
SendEmployedBees()	5.429
CalculateFitness()	0.132
main()	0.181
CalculateProbabilities()	0.179
MemorizeBestSource()	0.088
SendScoutBees()	0.032
init()	0.021

3.2 蜂群并行算法的实现

3.2.1 主从式蜂群并行算法框架图

采用不改变蜂群算法基本结构, 保持只有一个种群的主从式并行模型. 多核环境下蜂群并行算法框架图如图 1 所示.

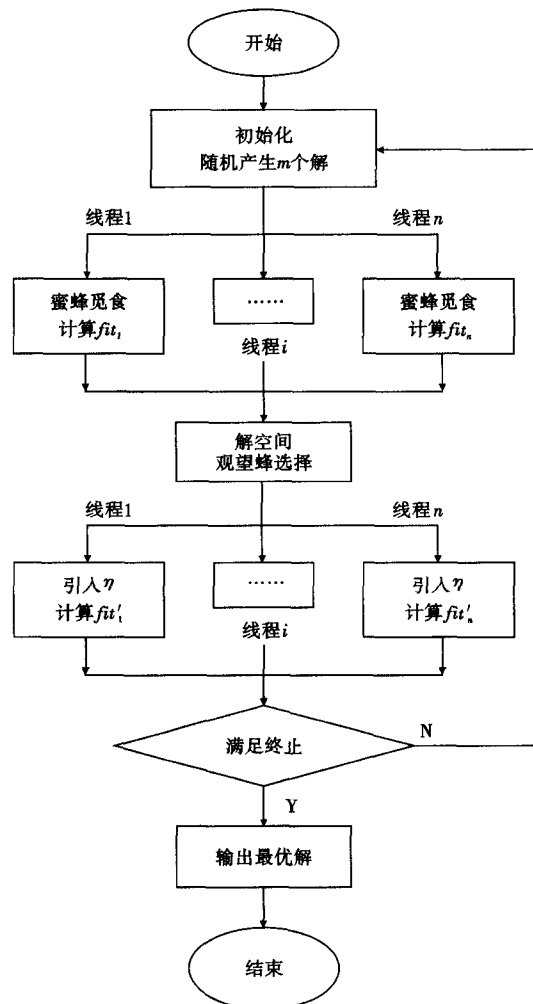


图 1 主从式蜂群算法框架图

3.2.2 蜂群并行算法的具体实现

并行部分都由 n 个线程来执行, n 的大小借助 OpenMP 中的 num_threads 字句动态改变. 并且用 OpenMP 中的 schedule 任务子句灵活调度任务, 达到各个线程间的负载均衡. 蜂群多线程并行算法主要实现步骤如下:

Step1: 初始化参数, 评估初始食物源的 fit 值;

Step2: 进入并行区, 自动将引领蜂分配到从处理器上运行;

Step3: 退出并行, 按照 fit 值对食物源排名;

Step4: 观望蜂选取食物源 x_i , 进入并行区, 每只观望蜂依公式(4)对解 x_i 产生新解 v_i ;

Step5: 退出并行, 选取 t 最大的引领蜂, if $t > limit$ 由式(3)随机产生的新解将其替换;

Step6: 记录到目前为止最好的解;

Step7: 比较迭代次数 cycle 与 maxCycle, if (cycle > maxCycle) 则输出最优解, 否则转(2).

4 实验及分析

4.1 实验平台及参数

实验硬件环境为 AMD Athlon(tm)64 位酷睿双核 PC, 内存 4G; 软件环境为 Microsoft Windows XP Professional (5.1, 2600); 编译环境为 Visual Studio 2005 (OpenMP), 性能测试工具为英特尔 Parallel Studio. 仿真实验中, $SN=20$, $BN=SN/2$, 取 $D=50$ 维与 $D=100$ 维, $maxCycle=2500$, $limit=100$, 测试函数分别独立执行 30 次求其平均值. 实验从文献[6]中选取三个基准函数进行对比测试.

(1) Sphere 函数:

$$f(x) = \sum_{i=1}^n x_i^2, -100 \leq x_i \leq 100$$

是连续的单峰函数, 在 $x_i=0 (i=1, 2, \dots, n)$ 时达到最小值 0.

(2) Griewank 函数:

$$f(x) = \frac{1}{400} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$$

$$-600 \leq x_i \leq 600$$

是复杂的非线性多峰值函数, 在 $x_i=0 (i=1, 2, \dots, n)$ 时达到最小值 0.

(3) Rastrigin 函数:

$$f(x) = \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i) + 10)$$

$$-5.12 \leq x_i \leq 5.12$$

是复杂的非线性多峰值函数, 在 $x_i=0 (i=1, 2, \dots,$

$n)$ 时达到最小值 0.

4.2 实验分析

分析一: 从表 2、表 3 可以看出, RP-ABC 算法在求函数的方差、最大值、最小值方面均优于 ABC 算法, 这充分说明了改进的新算法具有良好的全局搜索性能.

表 2 测试函数优化结果比较 ($D=50$)

函数	算法	方差	最大值	最小值
Sphere	ABC	1.24973E-13	4.21825E-13	2.6308E-15
	RP-ABC	2.83156E-16	2.09483E-15	1.0321E-15
	PRP-ABC	4.24732E-17	2.37654E-15	3.01425E-16
Griewank	ABC	1.73325E-02	9.26400E-2	6.2477E-15
	RP-ABC	2.14697E-09	9.34008E-06	0.0073E-16
	PRP-ABC	3.45854E-10	5.48962E-06	0.0051E-16
Rastrigin	ABC	0.46074	1.76652	4.7125E-12
	RP-ABC	8.74712E-10	4.09921E-09	6.1497E-14
	PRP-ABC	6.37524E-12	2.34528E-09	4.2548E-11

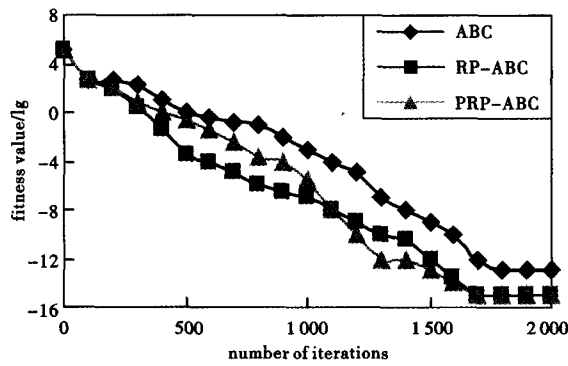
表 3 测试函数优化结果比较 ($D=100$)

函数	算法	方差	最大值	最小值
Sphere	ABC	6.09423E-05	1.73493E-04	2.4776E-09
	RP-ABC	2.49364E-15	0.04632E-12	4.0781E-13
	PRP-ABC	2.501275E-15	0.09564E-13	4.65872E-15
Griewank	ABC	4.68713	27.06907	8.64776
	RP-ABC	0.79112E-08	2.98234E-05	0.7422E-15
	PRP-ABC	0.54821E-11	4.24598E-08	0.5412E-17
Rastrigin	ABC	0.00772	0.02930	6.4765E-07
	RP-ABC	7.01312E-11	0.00973E-10	0.0004E-15
	PRP-ABC	6.54897E-16	0.00034E-10	0.0001E-20

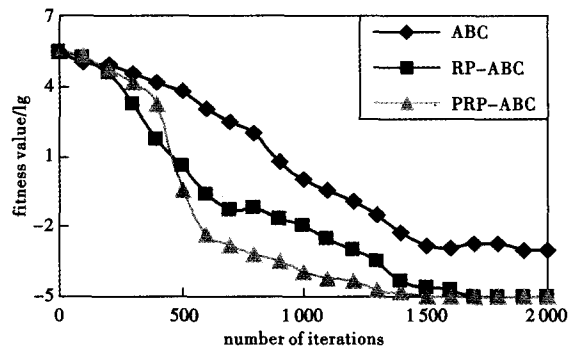
分析二: 从图 2 可以看出对于单模函数 Sphere, RP-ABC 算法略好于 ABC 算法, RP-ABC 在 50 维与 PRP-ABC 算法的收敛速度接近, 体现不出并行效率, 但在高维复杂空间下 PRP-ABC 算法的优势就显现出来.

分析三: 从图 3(a) 和图 4(a) 得出 RP-ABC 算法的收敛速度和精度均明显高于 ABC 算法, 并且在后期趋于稳定. PRP-ABC 算法在进化初期收敛速度略慢, 但是在后期收敛速度明显优于前两个算法. 从图 3(b) 和图 4(b) 中得出在高维复杂空间下 ABC 算法收敛非常慢, 几乎没有太多的变化, 而 RP-ABC 算法和 PRP-ABC 算法则可以快速搜索到最优值, 其中 PRP-ABC 算法在后期进化中的寻优速度和稳定性明显提高, Rastrigin 函数尤为明显.

分析四: 假若 RP-ABC 算法每次的平均运行时

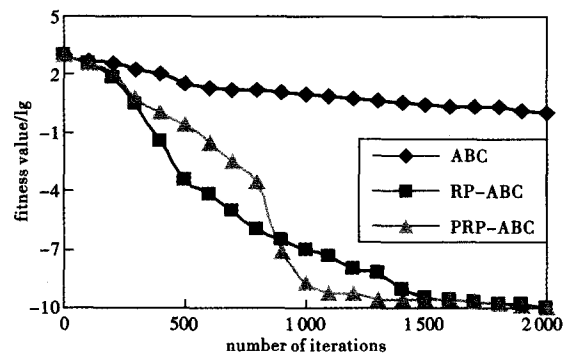


(a) D=50

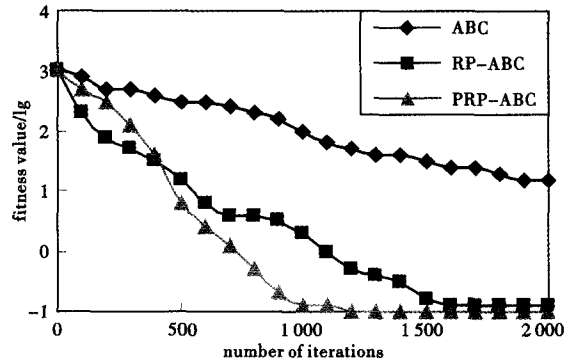


(b) D=100

图2 Sphere 函数不同维度对比

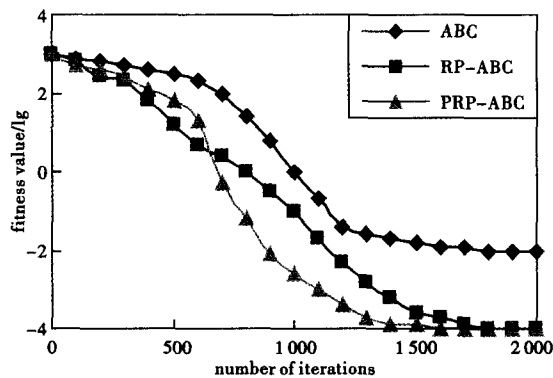


(a) D=50

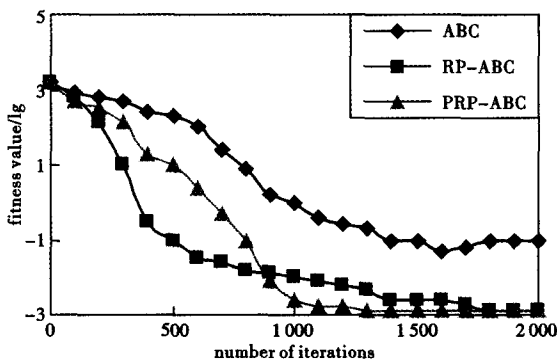


(b) D=100

图4 Rastrigin 函数不同维度对比



(a) D=50



(b) D=100

图3 Griewank 函数不同维度对比

间为 T_s , PRP-ABC 算法的每次平均运行时间为 T_p , 则加速比 S_p 为: $S_p = T_s / T_p$. 表 4 是两种算在 100 维下加速比, 可得出采用多核 PC 下的并行算法有效提高了计算性能和加速比性能, 其中 Griewank 函数提高了 80% 多, 其他两个算法也提高了近 70%, 这足以说明多核并行计算在求解搜索空间大的高维全局问题时的优越性.

表 4 算法在单核与双核上运行效率

函数名称	运行时间(D=100)		加速比
	ABC	PRP-ABC	
Sphere	0.00355	0.00130	1.667
Griewank	0.00643	0.00348	1.847
Rastrigin	0.00721	0.00425	1.696

5 结束语

本文在传统算法的基础上引入了随机摄动因子和全局最优解策略, 采用主从式并行模型, 实现了在单机多核下的人工蜂群并行算法. 通过仿真实验结果表明该算法具有较强的鲁棒性, 为高维复杂函数

(下转第 70 页)

5 结束语

本文针对 DV-Hop 算法误差较大的问题,对平均每跳距离做了相应改进,使平均跳距与实际平均跳距更加接近。实验结果表明,本文改进算法的定位误差明显小于原算法,定位精度有了很大改善。本文算法在文献[5]的基础上增加了一次数据广播,算法的通信量有所增加。因此,降低算法的通信量并保证定位精度是未来研究工作的重点。

参考文献:

- [1] He Tian, Huang Chengdu, Brian M BLUM, et al. Range-free localization schemes for large scale sensor networks[C]//Proc of the 9th Annual Int'l Conf on Mobile Computing and Networking. San Diego: ACM Press, 2003: 81-95.
- [2] Dragos Niculescu, Badri Nath. Ad hoc positioning system (APS) using AoA [C]//Proc of the IEEE INFOCOM 2003. USA: San Francisco, IEEE Computer and Communications Societies, 2003(3): 1734-1743.
- [3] Chen Long, Ahn Saeyoung, An Sunshin. An improved localization algorithm based on DV-Hop for Wireless Sensor Network [C]//IT Convergence and

Services. Atlanta, 2011:333-341.

- [4] Cui Xiaoyan, Hu Ting, Guo Mingjie. Improved DV-Hop algorithm based on self-localization of WSN[C]//The 1st International Conference on Information Science and Engineering. China: NanJing, 2009:4026-4029.
- [5] 彭刚,曹元大,孙利民. 无线传感器网络节点定位机制的研究[J]. 计算机工程与应用, 2004, 40(35): 27-29.
- [6] 范敏,胡伟. 一种新的无线传感器网络定位算法研究[J]. 微电子学与计算机, 2012, 29(2): 138-141.
- [7] 林今朝,刘海波,李国军,等. 无线传感器网络中 DV-Hop 节点定位改进算法研究[J]. 计算机应用研究, 2009, 26(4): 1272-1275.
- [8] 沈明玉,张寅. 基于改进的平均跳距和估计距离的 DV-Hop 定位算法[J]. 计算机应用研究, 2011, 28(2): 648-650.

作者简介:

苏兵男, (1972—), 博士, 副教授. 研究方向为 计算机软件工程、无线传感器网络、计算机网络及网络安全。

薛伟杰女, (1988—), 硕士研究生. 研究方向为无线传感器网络。

王洪元男, (1960—), 博士, 教授. 研究方向为模式识别与智能系统。

(上接第 66 页)

的优化问题提供了一种有效手段。下一步需要研究:

(1)该并行算法中初始化种群的规模大小、最大循环次数 limit 的取值对算法的影响;(2)为串行算法寻找其他更好的优化方法以及使并行算法具有更高的效率。

参考文献:

- [1] Karaboga D. An idea based on bee swarm for numerical optimization, TR06 [R]. [S. l.]: Erciyes University, Engineering Faculty, Computer Engineering Department, 2005.
- [2] 胡珂,李迅波,王振林. 改进的蜂群算法性能[J]. 计算机应用, 2011, 31(4): 1107-1110.
- [3] 孙晓雅,林焰. 改进的人工蜂群算法求解任务指派问题[J]. 微电子学与计算机, 2012, 29(1): 23-26.

[4] 李海生. 一类基于蜜蜂采集模型的智能算法[J]. 计算机与现代化, 2010(1): 7-11.

[5] 余柄辉,袁晓辉,王金文,等. 随机摄动粒子群优化算法[J]. 计算机工程, 2006, 32(12): 189-190.

[6] Karaboga D, Basturk B. On the performance of Artificial Bee Colony (ABC) algorithm [J]. Applied Soft Computing, 2008, 8(1): 687-697.

作者简介:

李双女, (1988—), 硕士研究生. 研究方向为并行计算、Petri 网。

李文敬男, (1964—), 教授. 研究方向为并行算法。

杨文男, (1987—), 硕士研究生. 研究方向为并行算法、多核程序设计。

周海燕女, (1985—), 硕士研究生. 研究方向为数据挖掘。