

文章编号:1001-9081(2005)02-0374-0

## 一种基于学习机制的并行遗传算法

张桂娟,武兆慧,刘希玉

(山东师范大学 信息管理学院,山东 济南 250014)

(guijuanzhang@hotmail.com)

**摘 要:**基于生物学群落的概念,提出了一个群落—种群—个体的三层模型,并在该模型上发展了一种基于学习机制的并行遗传算法(PGABL)。算法引入黑板模型作为控制和交互的数据结构,采用群内、群间、群落三个学习算子,将遗传进化和遗传学习相结合,有效地改善了遗传算法的性能。实验结果表明,该算法具有良好的适应性和稳定性。

**关键词:**遗传学习;早熟收敛;并行遗传算法

**中图分类号:**TP301.6 **文献标识码:**A

## Parallel genetic algorithm based on learning mechanism

ZHANG Gui-juan, WU Zhao-hui, LIU Xi-yu

(College of Information Management, Shandong Normal University, Shandong Jinan 250014, China)

**Abstract:** Based on the concept of biotic community in Biology, a 3-layer model named community-population-individual was proposed. Meanwhile, a parallel genetic algorithm based on learning mechanism (PGABL) was developed on this model. As the data structure for collaborations between subpopulations, the Blackboard model was introduced. And three learning operators are designed, through which PGABL combines the advantages of genetic evolution and genetic learning that improves the performance of traditional genetic algorithm effectively. Experimental results show that PGABL is of good adaptability and stability.

**Key words:** genetic learning; premature convergence; parallel genetic algorithm

### 0 引言

遗传算法 GA 是借鉴生物界自然选择思想和进化机制发展起来的高度并行的全局搜索算法,自提出以来在优化理论、数值分析等领域得到大量有效的应用<sup>[1]</sup>。但它在应用中有两个严重的缺点,即收敛速度慢和存在早熟收敛<sup>[2]</sup>。对此有些学者提出用自适应变化的遗传算子<sup>[3]</sup>、引入算子<sup>[4]</sup>等方法解决,取得了一定的成效。尽管如此,串行 GA 在速度、局部收敛以及大规模复杂问题的处理方面,仍存在着一些不足。近年来,随着计算机硬件技术的不断提高,遗传算法固有的并行性得到有效的利用,并行遗传算法的研究使得生物种群集体行为的复杂性在进化模拟中得到进一步的发展,在提高运算速度的同时,有效地维持了种群的多样性,因此成为遗传算法研究的新焦点之一<sup>[5]</sup>。目前已有三类较为成熟的并行模型:主从式模型、粗粒度模型和细粒度模型。

本文模拟生物学中群落的概念,在群落—种群—个体三层模型的基础上发展了一种基于学习机制的并行遗传算法 PGABL (Parallel Genetic Algorithm Based On Learning Mechanism)。以黑板模型作为算法控制和种群交互的数据结构,并引入三个学习算子实现学习机制。其中前两个学习算子充分利用了优秀模式的引导作用,加快了算法收敛的速度;

后一个学习算子通过对基因缺失位的补偿,保持了种群的多样性,有效地防止了早熟收敛。

### 1 相关定义

**定义 1** 群落: 群落  $C = \{P_1, P_2, \dots, P_n\}$  是所有种群的集合。

**定义 2** 年龄:  $\forall I \in P$ , 设  $I.age_i$  是个体  $I$  在第  $i$  代的年龄, 第  $i+1$  代的年龄为:  $I.age_{i+1} = I.age_i + \alpha \left| \frac{P.AvgFitness}{I.fitness} \right|$ ,  $\alpha$  为正常数。适应度高的个体老化速度慢, 会有更多的机会参与遗传操作。当个体年龄超过生命期时, 该个体死亡。

**定义 3** 模式抽取<sup>[6]</sup>: 设染色体  $e = (e_1, e_2, \dots, e_n)$ ,  $d = (d_1, d_2, \dots, d_n)$ ,  $e_i, d_i \in \{0, 1, *\}$ 。定义  $e \times d = g, g_i = \begin{cases} e_i, & e_i = d_i \\ *, & \text{其他} \end{cases}$ , 此时  $g$  所代表的模式为染色体  $e$  与  $d$  的祖先模式。模式抽取即为获取已知染色体祖先模式的过程。

**定义 4** 种群模式:  $\forall p \in C, P = \{I_1, I_2, \dots, I_n\}$ ,  $P.LocalSchema = I_1 \times I_2 \times \dots \times I_n$ 。

**定义 5** 优秀率:  $\forall p \in C$ , 记  $P.excratio = c * \lfloor P.AvgFitness / P.BestFitness \rfloor$ ,  $c$  为正常数。在没有大的偏差的情况下, 种群平均适应度与最优个体适应度的比值越大说明种

收稿日期:2004-07-27;修订日期:2004-09-29

基金项目:国家自然科学基金资助项目(6037405);山东省自然科学基金重大项目(Z2004G02);山东省中青年科学家奖励基金资助项目(03BS003)

作者简介:张桂娟(1981-),女,山东日照人,硕士研究生,主要研究方向:进化计算;武兆慧(1981-),女,内蒙古包头人,硕士研究生,主要研究方向:遗传算法、数据挖掘;刘希玉(1964-),男,山东济南人,教授,博士,博士生导师,主要研究方向:进化计算、人工神经网络。

群中优秀个体越多,优秀率越高。

**定义6** 种群最优模式:  $\forall p \in C$ , 种群最优模式  $P_{\text{BestSchema}} = I_{m1} \times I_{m2} \times \dots \times I_{mr}$ , 其中  $r = P_{\text{Size}} * p_{\text{excratio}}$ ,  $\text{Min}(I_{m_i}.\text{fitness}) \geq \text{Max}(I_{m_t}.\text{fitness}), i = 1, 2, \dots, r, t = r + 1, \dots, P_{\text{Size}}$

**定义7** 全局最优模式: 设群落  $C = \{P_1, P_2, \dots, P_n\}$ , 则  $C_{\text{BestSchema}} = P_1_{\text{BestSchema}} \times P_2_{\text{BestSchema}} \times \dots \times P_n_{\text{BestSchema}}$ 。

**定义8** 模式学习: 设染色体  $e = (e_1, e_2, \dots, e_n)$ ,  $e_i \in \{0, 1\}$ , 模式  $d = (d_1, d_2, \dots, d_n)$ ,  $d_i \in \{0, 1, *\}$ 。定义染色体  $e$  向模式  $d$  学习,  $e \odot d = g$ ,  $g_i = \begin{cases} e_i, & e_i = d_i \text{ 或者 } d_i = * \\ 1 - e_i, & \text{其他} \end{cases}$ 。

**定义9** 基因补偿:  $\forall p \in C, P$  的种群模式  $P_{\text{LocalSchema}} = (x_1, x_2, \dots, x_n)$ , 如果对于  $\forall i \in \{1, 2, \dots, n\}$ , 只要  $x_i \neq *$ , 则第  $i$  位发生基因丢失。从种群中选择一定数量的较差个体, 将第  $i$  位翻转称为基因补偿。

## 2 算法中涉及的几个关键问题

### 2.1 种群的划分

多种群遗传算法种群的划分策略大多采用随机方法, 子种群个数一般预设为结点机个数, 这种划分方法不能充分利用个体之间的信息差别。本文借鉴自适应聚类划分方法, 以海明距离作为测度标准, 使得各种群的初始群体尽量分布在整个可行域的不同区域中, 在保持群体的多样性的同时也提高了算法的局部搜索性能。设网络中结点机个数  $node$ , 算法流程如下:

1) 令  $C = \{I_1, I_2, I_3, \dots, I_N\}$ , 代表一个含有  $N$  个个体的群落;

2) 令  $H = \Phi$ ;

3) 从  $C$  中任选个体  $I_j$ , 并将  $I_j$  加入  $H_1$  中,  $I_j \rightarrow H_1$ ;

4) 在  $C$  中选择个体  $I_i$ , 使得  $D(I_i - I_j)$  最大,  $I_i \rightarrow H_2$ ;

5)  $H \leftarrow H_1 \cup H_2, k = 2$ ;

6) 对于  $\forall I_j \in C$ , 并且  $I_j \notin H$ , 将它们分配到  $k$  个类中  $\{S_1, S_2, \dots, S_k\}$ , 如果对于  $i = 1, 2, \dots, k, p \in \{1, 2, \dots, k\}$ ,  $D(H_p - I_j) = \text{Min}(D(H_i - I_j))$ , 则  $I_j \in S_p$ ;

7) 对于  $i, t = 1, 2, \dots, k$ , 如果  $\text{Max}(D(H_i - I_j)) > \text{Avg}(D(H_i - H_t))$ , 则  $k = k + 1, H_k = I_j, H \leftarrow H \cup H_k$ , 转6), 否则转8);

8) if  $k \leq node$ , 转9), 否则, 执行下面的操作:

选择  $H_p, H_j (p, j \in \{1, 2, \dots, k\})$ , 使得  $D(H_p - H_j) = \text{Min}(D(H_i - H_t)) (i, t = 1, 2, \dots, k)$ , 合并  $S_p, S_j, H_j$ , 中心设为  $H_p, k = k - 1$ , 转8);

9)  $H_i \leftarrow H_i \cup S_i, i = 1, 2, \dots, k$ 。

这样整个群落  $C$  就被划分为若干种群  $H_i$ 。种群之间通过黑板模型交互, 相互学习和竞争, 以达到并行进化的目的。

### 2.2 黑板模型及功能特征

黑板是协作时常采用的方法, 该模型已被广泛使用, 其相应的技术也较为成熟<sup>[7]</sup>。本文利用黑板模型作为整个群落交互的数据结构和算法的控制中心, 在算法实现过程中, 以主进程的形式出现, 它需要处理的任务有:

1) 随时接收各种群的最优个体和种群最优模式。

2) 比较并保存全局最优个体。

3) 根据种群最优模式, 抽取全局最优模式。

4) 向发出群间学习请求的种群发送全局最优模式。

5) 若只剩一种群未终止, 停止群间学习。

6) 若全部群体均终止则算法结束, 输出黑板上的最优解。

### 2.3 学习机制

为实现学习机制, 算法引入群内、群间和群落三个学习算子。其中, 群内和群间学习算子是通过种群个体向最优模式的学习来加快算法的收敛速度, 充分发挥优秀模式的引导作用; 群落学习算子则是通过对种群模式基因丢失位的基因补偿来防止早熟现象的发生。因此三个学习算子的引入可以有效地克服传统遗传算法的两个缺点, 改善遗传算法的搜索性能。

#### 2.3.1 群内学习算子

群内学习算子是通过控制种群中的一部分较差个体对种群(局部)最优模式进行学习来实现的。设待学习个体  $I$  的染色体  $e = 00110001$ , 种群最优模式  $d = 10 * 0 * * 1 *$ ,  $e \odot d = g$ , 易得  $g = 10100011$ 。群内学习需要解决的一个关键问题是群内学习频率的确定。算法开始, 为加快种群的进化速度, 学习频率稍大; 当算法慢慢收敛时, 为了保持种群的多样性, 学习频率逐渐减小。

$$P_{\text{IntStuRatio}} = \lambda | P_{\text{AvgFitness}} / P_{\text{BestFitness}} | \quad (1)$$

下面是群内学习操作流程:

1) 生成  $[0, 1]$  之间的随机数  $ran$ , 如果  $ran < P_{\text{IntStuRatio}}$ , 则执行下面的操作;

2) 计算种群的优秀率和优秀个体的数目  $m$ ;

3) 按照个体适应度选择  $m$  个优秀个体;

4) 对这些优秀个体进行基因抽取获取种群最优模式;

5) 随机选择若干种内最差个体对最优模式进行学习, 及时更新个体适应度;

6) 将种群最优模式及最优个体提交给黑板。

#### 2.3.2 群间学习算子

群间学习算子是通过控制种群内部部分个体对全局最优模式的学习来实现的。它类似于孤岛模型中的个体迁移策略, 所不同的是, 这里迁移的并非个体, 而是全局最优模式。通过引入随机个体对全局最优模式进行学习, 以确定全局最优模式的非确定位, 实现对全局最优模式的局部细搜, 同时也为种群吸纳了优秀的基因, 加快了算法的收敛进程。考虑到算法本身的仿生学特征, 利用生物学的观点解决群间学习频率的选取问题。当一个种群的平均适应度较低时, 该种群为了赶超其他种群, 应更有必要向优秀模式学习, 因此设定低平均适应度的种群的群间学习频率高。

$$P_{\text{MidStrRatio}} =$$

$$\mu_j \left| \frac{\text{Max}(P_i.\text{AvgFitness}) - P_j.\text{AvgFitness}}{\text{Max}(P_i.\text{AvgFitness}) - \text{Min}(P_i.\text{AvgFitness})} \right| \quad (2)$$

$i = 1, 2, \dots, C_{\text{Size}}, j = 1, 2, \dots, C_{\text{Size}}, \mu_j$  为调整参数, 取值视具体问题而定, 一般取 1.0。针对每个种群实施下面的群间学习操作:

1) 生成  $[0, 1]$  之间的随机数  $ran$ , 如果  $ran < P_{\text{MidStrRatio}}$ ,

MidStuRatio, 则执行下面的操作;

2) 向黑板发送群间学习请求, 在可能的情况下获取黑板上的全局最优模式;

3) 随机生成若干个体, 并对全局最优模式进行学习, 计算个体的适应度, 并置年龄为 0。

### 2.3.3 群落学习算子

群落学习算子是对种群基因缺失位的补偿。例如设种群  $P$ . LocalSchema = \* \* 0 \* 1 \* \* \*, 基因缺失位为第 3 位和第 5 位。这样就可以从种群  $P$  中随机选择若干最差个体将第 3 位和第 5 位翻转即可。群落学习的频率与群内学习频率成反比。算法开始时较小, 而随着算法的进行, 为防止早熟收敛和保持种群多样性, 群落学习频率不断增大。对每个种群, 算子操作流程如下:

1) 生成  $[0, 1]$  之间的随机数  $ran$ , 如果  $ran < P$ . ComStuRatio, 则执行下面的操作;

2) 对种群进行基因抽取, 获取种群基因模式;

3) 对丢失的基因位进行基因补偿。

## 3 算法描述

该算法首先随机生成一定数量的个体组成一个群落, 然后利用上述方法划分种群。划分后的各种群在进化的过程中, 一方面进行正常的遗传操作, 计算并更新个体的年龄, 将最优个体的副本传递给黑板; 另一方面, 需要按一定的频率进行群内、群间和群落学习。当种群进行群间学习时, 该种群首先向黑板发送群间学习的请求, 当接收到群间学习请求并在条件允许的情况下, 黑板就会将当前的最新信息发送至发出请求的种群。算法在达到要求的收敛精度时停止, 具体流程如下:

1) 产生一个进程(称为黑板进程)并初始化参数;

2) 黑板进程随机生成一定数量的个体, 划分种群, 产生与种群数量相等的  $k$  个进程, 以实现各个种群的独立进化;

3) 每个种群单独进化:

a) 选择、交叉、变异操作, 修改个体适应度及年龄;

b) 群内、群间、群落学习;

c) 检查终止条件, 若满足转 d), 否则转 a);

d) 种群停止进化, 通知黑板并提交最优解。

4) 黑板进程的任务处理。

## 4 仿真试验分析

本文采用的环境是主从式控制网络并行 GA, 在可移植的编程环境 PVM(Parallel Virtual Machine)下实现的。PVM 系统是一个能使异构的计算机群体作为一个紧凑、灵活、经济的计算机资源来使用的并行算法环境。它支持 TCP/IP 网络协议, 可以安装在多种操作系统上, 如 Unix、Linux、Windows, 目前允许用户使用 C、C++ 或 Fortran 编程, 通过调用 PVM 库函数来进行进程调度、虚拟机控制以及任务间的通信等。其特点是通用性强(只要求网络支持 TCP/IP), 系统规模小, 技术比较成熟。我们采用纯节点编程模式, 将黑板模型作为主进程, 其他子群体作为子进程。由于黑板具有较多的通信负担, 应该选择一个合适档次的处理机。选取两个典型测试函数对 PGABL 的性能进行测试。

F1: Camel 函数。具有 6 个局部极小点, 两个为全局最小

点, 最小值为  $-1.031628$ 。

$$f_1(x, y) = (4 - 2.1x^2 + x^4/3)x^2 + xy + (-4 + 4y^2)y^2, \quad -10 < x, y < 10 \quad (3)$$

F2: Rosenbrock 函数。有两个局部极大点  $f(2.048, -2.048) = 3897.7342$  和  $f(-2.048, -2.048) = 3905.926$ , 其中后者为全局最大, 用遗传算法对其进行优化计算时极易陷入前一个局部极大点。

$$\begin{cases} \max f(x_1, x_2) = 100(x_1^2 - x_2^2)^2 + (1 - x_1^2) \\ s. t. \quad -2.048 \leq x_i \leq 2.048, i = 1, 2 \end{cases} \quad (4)$$

PGA1 是基于孤岛模型的遗传算法 PGA, 其各个子进程采用传统遗传算法 SGA, 子种群规模为 20, 种群个数为 5, 迁移间隔取为 3 的倍数。对于 PGABL 的群落个体数为 100, 年龄控制  $a = 1$ ,  $life = 50$ , 优秀率  $c = 1$ 。两个算法均取交叉率  $pm = 0.80$ , 变异率  $pc = 0.06$ , 实验共运行 50 次, 每次设最大进化代数 100。比较结果如表 1 所示。

由表 1 可以看出, 引入学习机制的 PGABL 比采用 SGA 的一般孤岛模型具有更高的收敛速度和收敛精度。在 50 次试验中 PGABL 几乎每次都能收敛到全局最优解, 或全局近似最优解。

表 1 PGABL 与 PGA 算法性能比较

函数	算法	指定代数内收敛次数			效率	最优解
		0 ~ 30	30 ~ 60	60 ~ 100		
F1	PGA1	8	18	20	0.92	-1.0316279
	PGABL	12	22	14	0.96	-1.0316280
F2	PGA1	10	29	2	0.82	3905.92604
	PGABL	16	28	0	0.88	3905.92602

## 5 结语

本文提出的基于学习机制并行遗传算法 (PGABL) 将进化和学习相结合, 有效地改善了遗传算法的搜索性能。三个学习算子不仅加速了算法的收敛速度, 还有效防止了早熟现象。实验结果也正说明了这一点。算法内部还有一部分控制参数的选取是依据经验而定的, 下一步改进的目标是进一步提高算法的自适应能力。

### 参考文献:

- [1] BUCKLES BP, PETRY FE. Genetic Algorithms [M]. Los Alamitos, Calif: IEEE Computer Society Press, 1992.
- [2] 潘正君, 康立山, 陈毓屏. 演化计算 [M]. 北京: 清华大学出版社: 南宁: 广西科学技术出版社, 1998.
- [3] SRINIVAS M, PATNAIK LM. Adaptive Probabilities of Crossover and Mutation in Genetic Algorithms [J]. IEEE Trans on Systems, Man and Cybernetics, 1994, 24(4): 656 ~ 667.
- [4] JONES T. Crossover, MacroMutation and Population - Based Search [A]. In Eshelman Led: Proceedings of the 6th international conference on Genetic Algorithms [C], San Francisco, CA: Morgan Kaufmann, 1995. 73 ~ 80.
- [5] COLLINS RJ, JEFFERSON DR. Selection in Massively parallel genetic algorithm [A]. Proc Of the fourth ICGA [C], 1993. 249 ~ 256.
- [6] 熊伟清, 魏平, 赵志煜. 遗传算法的一个调节算子的研究 [J]. 小型微型计算机系统, 2003, 24(3): 531 ~ 533.
- [7] 刘弘, 林宗楷. 一种支持动态任务分配的协同设计方法 [J]. 软件学报, 2001, 12(12): 1830 ~ 1833.