

# 基于免疫量子进化算法的负载均衡策略

苏日娜, 王 宇

(宁波工程学院电子与信息工程学院, 浙江 宁波 315016)

**摘 要:** 在集群系统任务调度和分配中, 提出一种基于免疫量子进化算法的负载均衡策略。该策略采用量子化编码和量子进化操作优化任务分配, 在量子陷入局部极值下, 引入免疫操作进行接种疫苗和免疫选择, 从而增加种群多样性。仿真结果表明, 与 SGALB 策略相比, 该策略具有更高的搜索效率, 其集群系统的整体性能更优。

**关键词:** 免疫量子进化算法; 集群系统; 负载均衡; 调度器

## Load Balancing Strategy Based on Immune Quantum Evolutionary Algorithm

SU Ri-na, WANG Yu

(College of Electronic and Information Engineering, Ningbo University of Technology, Ningbo 315016, China)

**【Abstract】** In the task scheduling and allocation of cluster system, this paper presents a load balancing strategy based on Immune Quantum Evolutionary Algorithm(IQEA). This strategy optimizes load distribution by quantum coding and quantum evolution operator. It ensures the diversity of population by using immune operator vaccinations and immune selection when quantum is into the local extremum. Simulation result shows that this strategy has higher search efficiency than the SGALB strategy, and it can improve the whole performance of cluster system effectively.

**【Key words】** Immune Quantum Evolutionary Algorithm(IQEA); cluster system; load balancing; scheduler

DOI: 10.3969/j.issn.1000-3428.2011.02.053

### 1 概述

网络的快速发展和普及使得对计算机系统服务器的访问量日益增加, 单个服务器已经不能满足对系统可伸缩性和可用性的要求。目前, 通常采用构建集群环境以解决用户访问量增加而导致的系统性能下降问题。影响集群系统性能好坏的关键是如何合理地将任务分配到各个服务器, 实现最优化的负载调度, 即负载均衡。负载均衡的主要目的是寻求任务请求和服务资源的最佳匹配方案。根据各个节点服务器当前负载量的高低和处理能力的不同, 对负载重的服务器分配较多的任务, 对负载轻的服务器分配较少的任务, 对于故障的节点服务器则将任务交由负载最轻的服务器处理, 从而平衡系统。研究证明, 多任务调度和分配问题是一个 NP 问题。目前, 传统负载均衡算法主要分为静态负载均衡和动态负载均衡<sup>[1]2</sup> 大类, 很多文献都对此进行了相关研究, 各种算法都有相应的优缺点<sup>[2-3]</sup>。随着优化技术的发展, 许多智能优化方法被成功运用到解决负载均衡这个 NP 难问题。例如, 基于遗传算法的负载均衡<sup>[4-5]</sup>(SGALB)和基于群智能算法的负载均衡方法<sup>[6]</sup>。

量子进化计算是近年来提出的一种将量子计算与进化计算相融合的新型优化算法。该算法相比传统遗传算法具有更好的种群多样性并有效克服了传统遗传算法的早熟现象, 且比群智能算法求解精度更高。同时, 为进一步提高种群多样性, 采用免疫策略进行量子免疫突变, 对退化的量子个体进行疫苗接种和免疫选择, 使量子能够摆脱局部极值达到更优的进化效果。

因此, 结合当前集群系统面临的负载均衡和任务调度的实际问题, 本文将免疫量子进化算法(Immune Quantum Evolutionary Algorithm, IQEA)应用到负载均衡中, 在仿真集群系统

上进行测试实验, 并与 SGALB 策略进行对比, 实现了更高效的任务分配效率, 有效提高了系统的整体性能。

### 2 问题描述及数学模型

在集群系统中, 负载均衡问题可以描述如下: 有  $N$  个新的任务, 每个任务请求完成时间为  $t_i$ , 需要分配到  $M$  个负载量和处理能力各不相同的节点服务器上进行处理, 目标是找到一个最优的分配方案, 使整个任务处理的时间最短。

对上述问题描述建立数学模型, 假设  $M$  台服务器分别为  $S_1, S_2, \dots, S_M$ , 各自当前的负载量为  $l_1, l_2, \dots, l_M$ , 有  $N$  个需要分配的任务, 每个任务所需处理时间分别为  $t_1, t_2, \dots, t_N$ 。本文采用所有节点服务器上任务处理代价与该节点负载量的偏差总和最小表示系统处于最优状态, 数学表示为:

$$F(X) = P_{\min} = \sum_{i=1}^M w(n_i, l_i, q_i) \quad (1)$$

$$w(n_i, l_i, q_i) = C_1 \left( \sum_{j=1}^{n_i} t_j + q_i \bar{t}_i \right) - C_2 l_i \quad (2)$$

其中,  $X = \{n_1, n_2, \dots, n_M\}$  为一个分配方案;  $M$  是集群中的服务器节点数;  $N$  为待分配的总任务数;  $n_i$  为第  $i$  个工作节点上预分配的新任务数;  $l_i$  为第  $i$  个工作节点的当前负载;  $q_i$  为第  $i$  个工作节点服务器上就绪队列长度;  $\bar{t}_i$  为第  $i$  个工作节点平均处理时间;  $C_1, C_2$  为常数;  $w(n_i, l_i, q_i)$  是一个能反映所要分配任务与节点处理能力偏差的函数, 当节点处理的任务

**基金项目:** 浙江省自然科学基金资助项目(Y1080123); 浙江省教育厅基金资助项目(Y201016215)

**作者简介:** 苏日娜(1978 - ), 女, 讲师、硕士, 主研方向: 人工智能; 王 宇, 副教授、博士后

**收稿日期:** 2010-09-25 **E-mail:** sm2009@126.com

与节点负载能力最大限度匹配,则整个系统处于最佳运行状态,由此可以保证所有任务都得到最佳响应。同时,根据给出的目标函数,建立适应度函数为  $f(x) = \frac{1}{1+F(x)}$ , 目标函数

越小则适应性越大。本文将适应度函数的优化作为寻优的目标,节点服务器分配方案作为量子个体,采用免疫量子进化算法对量子进化寻优,待求的最优解是那些使得目标函数最小、适应度最高的量子。

### 3 量子进化算法

量子进化算法(QEA)<sup>[7]</sup>是基于量子比特和量子叠加态的概念和理论提出的一种新型优化算法。它采用量子比特进行个体编码,量子位是表示信息的最小单位,一位的信息位是由2个本征态 $|0\rangle$ 或 $|1\rangle$ 及其任意叠加态所构成的态变量,用概率幅表示为 $[\alpha_i, \beta_i]^T$ 且满足归一化条件 $|\alpha_i|^2 + |\beta_i|^2 = 1$ 。从而一个用 $n$ 个量子位编码的个体可以表示 $2^n$ 种状态。

量子进化采用量子旋转门作为进化策略,原理是使当前解逐渐逼近搜索到的最优解,将需要的结果以概率的形式增加,不需要的结果以概率的形式减弱,保证进化向最优方向进行。设算法进化到第 $t$ 代时,群体为 $P(t) = \{p_1^t, p_2^t, \dots, p_N^t\}$ ,对于量子 $p_j^t$ 的第 $i$ 个基因位 $[\alpha_i, \beta_i]^T$ ,量子旋转门进化为 $[\alpha_i', \beta_i']^T$ 的过程可以描述为:

$$\begin{bmatrix} \alpha_i' \\ \beta_i' \end{bmatrix} = \begin{bmatrix} \cos \theta_i & -\sin \theta_i \\ \sin \theta_i & \cos \theta_i \end{bmatrix} \begin{bmatrix} \alpha_i \\ \beta_i \end{bmatrix} \quad (3)$$

其中, $\theta_i$ 为旋转角,由下式计算:

$$\theta_i = \Delta \theta_i s(\alpha_i, \beta_i) \pi \quad (4)$$

其中, $\Delta \theta_i$ 为量子旋转门旋转角的大小,控制算法的收敛速度和性能;函数 $s(\alpha_i, \beta_i)$ 控制量子旋转门旋转角的方向,使算法向最优解方向搜索。一般情况下, $\Delta \theta_i$ 和 $s(\alpha_i, \beta_i)$ 是通过查表的方式进行计算。考虑到算法迭代初期应采用较大旋转角增加搜索范围以加快搜索速度,迭代后期采用较小旋转角缩小搜索范围以提高搜索精度,即搜索范围应当随着进化代数的增加而减少,因此,本文采用自适应方式调整旋转角度。旋转公式为:

$$\Delta \theta_i = 0.5\pi \cdot \text{rand}() \times (\text{iter} / t) \pi \quad (5)$$

其中, $\text{iter}$ 为当前迭代次数; $t$ 为总迭代次数。采用动态步长调整策略后,可以有效实现算法在速度和精度上的整体协调。

量子变异操作通常采用量子非门实现。算法按照变异概率对种群中所选个体的一个或若干个量子位进行突变,将对量子位的2个概率幅进行反转。量子变异操作实际上是改变了该量子比特态叠加的状态,使得原来朝向状态“1”方向坍塌,转变为朝向状态“0”方向坍塌,或者相反。

## 4 基于免疫量子进化算法的负载均衡策略

### 4.1 量子编码

免疫量子进化算法将一种服务器节点分配方案对应为一个量子染色体个体,采用多量子比特编码多状态基因<sup>[8]</sup>,初始产生具有 $N$ 个个体的种群 $Q(t) = (Q_1^t, Q_2^t, \dots, Q_j^t, \dots, Q_N^t)$ ,其中, $Q_j^t$ 为第 $t$ 次迭代的第 $j$ 个个体染色体,根据负载均衡问题描述,定义其长度为 $n \times m$ ,表示为:

$$Q_j^t = \begin{pmatrix} \alpha_{j1}^t & \alpha_{j2}^t & \dots & \alpha_{jnm}^t \\ \beta_{j1}^t & \beta_{j2}^t & \dots & \beta_{jnm}^t \end{pmatrix} (j=1, 2, \dots, N) \quad (6)$$

其中, $n$ 表示系统请求数量; $m$ 表示系统节点服务器个数,位置表示待分配任务序号;初始所有的 $\alpha_i$ 和 $\beta_i$ 都取值为

$1/\sqrt{2}$ 。经测量可以得到量子比特的观测态,具体操作为:随机产生一个 $[0, 1]$ 之间的随机数,若它大于 $|\alpha_i|^2$ ,则对应的二进制解的比特位取值1,否则取值0。因此通过对 $Q_j^t$ 测量可以得到一种可能状态解,即一种系统分配方案,采用长度为 $n \times m$ 的二进制串 $P(t) = (p_{11}^t, p_{21}^t, \dots, p_{ixj}^t, \dots, p_{nm}^t)$ 表示。其中,位置 $p_{ij}^t$ 上的值表示第 $t$ 次迭代请求 $i$ 是否分配给节点服务器 $j$ 来响应,是则为1,反之为0,从而可以根据该解的适应度值判断解的优劣。通过对问题进行量子化编码确保在同样数量种群个体情况下扩大搜索空间,从而增加种群的多样性。

### 4.2 免疫算子

为进一步增强种群多样性,避免量子在寻优中陷入局部极值,本文引入免疫算子对量子进行突变,主要包括接种疫苗和免疫检测。首先,种群经过前面的优化,假设最优解的个别基因片段已经出现在一些量子中,将它们当作疫苗抽取出来并接种到其他量子。假设每个量子经过测量后的二进制表示的每个基因位有 $s$ 个字符 $k_1, k_2, \dots, k_s$ 可供选择,在该种群中第 $i$ 等位基因上 $k_j$ 的概率为:

$$p_{ij} = \frac{1}{N} \sum_{j=1}^N a_j \quad (7)$$

其中, $a_j = \begin{cases} 1 & g(i) = k_j \\ 0 & \text{其他} \end{cases}$ ;  $g(i)$ 为种群中第 $i$ 等位基因上的符号; $N$ 为种群的个体数量。

将该等位基因上最大概率大于某个设定阈值 $T_b$ 的 $k_j$ 作为该等位基因上的疫苗片段,从而最终提取的疫苗为:

$$H = (h_1, h_2, \dots, h_n) \quad (8)$$

$$h_j = \begin{cases} k_j & \max(p_{ij}) > T_b \\ 0 & \text{其他} \end{cases} \quad (9)$$

然后,从父代群体中随机抽取个体,用生成的疫苗 $H$ 代替相应的基因位,形成新的种群。同时对接种疫苗后的个体进行检测和测量,如果适应度函数优于父代,则子代将代替父代进入下一代种群。如果适应度不如父代,则该个体将被父代中对应的个体替代。在目前的子代群体中以一定选择概率选择个体进入新的父代群体。

### 4.3 负载均衡的实现

本文首先根据集群系统服务器的具体负载情况和网络实际情况对网络结构进行预处理,对每个节点用 $1, 2, \dots, R-1, R$ 进行编号,并获取每个节点的负载量。然后实施负载均衡调度。下文给出实现的具体步骤:

**步骤1** 初始化种群 $Q(t) = (Q_1^t, Q_2^t, \dots, Q_j^t, \dots, Q_N^t)$  ( $j=1, 2, \dots, N$ ),初始化进化代数 $t=0$ 。根据量子比特编码方法,对初始服务器调度序列进行量子比特编码,生成 $N$ 个量子位为 $n \times m$ 的个体。所有量子位初始值设置为 $1/\sqrt{2}$ 。

**步骤2** 对种群 $Q(t)$ 的每个量子个体观测其叠加态以及各状态的概率值得到 $P(t)$ 。

**步骤3** 评估 $P(t)$ ,根据适应度值选择并保存当前最优解并将其制作为疫苗保存备用。

**步骤4** 对 $Q(t)$ 进行量子旋转门进化操作。

**步骤5** 对 $Q(t)$ 进行量子突变操作。

**步骤6** 判断种群多样性,如果条件不满足则表示陷入局部极值,则进行疫苗接种,对接种后的新一代进行检测并判断是否优于父代,优于则进行免疫选择,否则直接进行下一步。

**步骤7** 判定算法的停止条件,如满足停止条件则输出当

前个体,算法结束;否则  $t=t+1$ ,转步骤 2。停止条件主要是根据要求通过进化代数  $t$  确定。

### 5 仿真实验

在 Linux 平台上构建集群仿真环境。系统由 5 台服务器构成。1 台作为负载调度器在其上实施 IQEA 负载均衡策略。该调度器入口端用于接收客户端访问请求,出口端连接其他 4 台作为节点的服务器。在每个节点服务器上设计一个采集程序,定期将服务器的各项性能参数发送给负载调度器。负载调度器计算出当前每个节点的负载量,根据请求队列的要求进行任务分配。客户机模拟外部访问,用于测试集群系统的性能。经过对客户端测试脚本的编辑可调整每个发送请求的大小和数量。对系统进行测试,在迭代次数相同的情况下,分别采用 SGALB 和本文策略进行任务调度并测试集群系统性能。SGALB 和本文策略均采用终止进化代数为 600,种群规模  $N=50$ ,SGALB 中交叉概率为 0.7,变异概率为 0.4,本文策略中突变概率为 0.2。在实施 2 种负载方法下,通过测试软件工具 CHARIOT 对能体现集群性能优劣的 2 个指标,即系统完成请求的平均应答延时和数据吞吐量进行对比分析,测试结果如图 1、图 2 所示。

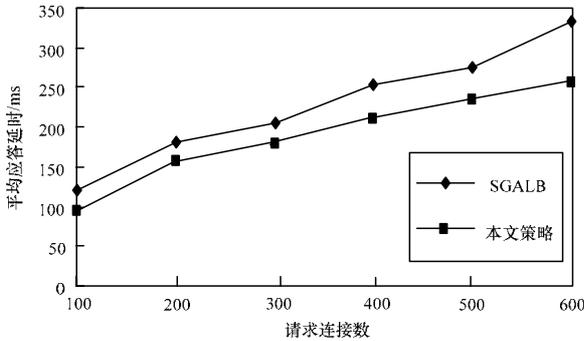


图 1 2 种策略的平均应答延时对比

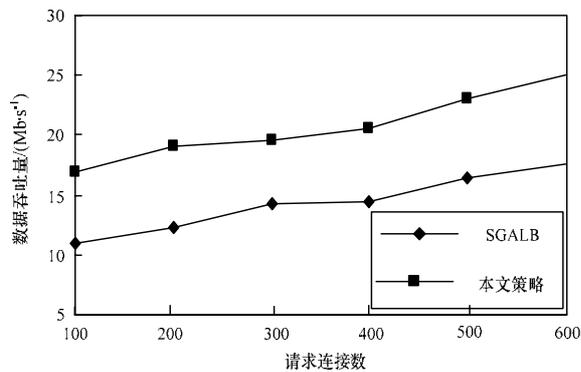


图 2 2 种策略的数据吞吐量对比

通过对比可以看出,与 SGALB 相比,本文策略能够有效降低平均应答延时、增加数据吞吐量。多次测试表明,在较少并发请求的情况下,IQEA 与 SGALB 的服务器性能差异较小,随着并发请求数量的增加,基于 IQEA 的服务器性能的优势越来越明显,并且丢包的情况明显减少。同时将本文

策略与 SGALB 以迭代次数对应适应度函数值的变化情况进行比较,结果如图 3 所示。

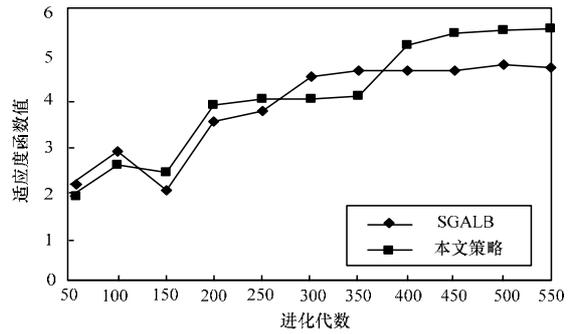


图 3 2 种策略的适应度函数值比较

从实验结果可以看出,相比 SGALB 策略,IQEA 能寻优到更好的解,搜索速度更快。而且本文策略由于其种群多样性增加,并结合陷入局部极值后的免疫操作,有效地避免了 SGALB 策略的早熟收敛现象。

### 6 结束语

本文提出一种基于免疫量子进化算法的负载均衡策略,在 SGALB 策略基础上对种群进行量子化编码和量子进化操作,引入免疫算子对量子进行突变,极大增强个体多样性,有效克服了 SGALB 策略的退化现象。结果表明,本文策略使整个系统具有较好的负载运行状态。

### 参考文献

- [1] 王春娟,董丽丽,贾丽. Web 集群系统的负载均衡算法[J]. 计算机工程, 2010, 36(2): 102-104.
- [2] Kim C, Kameda H. Optimal Static Load Balancing of Multi-class Jobs in a Distributed Computer System[C]//Proc. of the 10th International Conference on Distributed Computing Systems. Washington D. C., USA: IEEE Computer Society, 1990.
- [3] Cardellini V, Colajanni M. Dynamic Load Balancing on Web-server Systems[J]. IEEE Internet Computing, 1999, 3(3): 28-39.
- [4] Zomaya A Y. Observations on Using Genetic Algorithms for Dynamic Load-balancing[J]. IEEE Transactions on Parallel and Distributed Systems, 2001, 12(9): 899-911.
- [5] 张维勇,张华忠,柳楠. 基于遗传算法的服务器端负载均衡系统的设计[J]. 计算机工程, 2005, 31(20): 121-123.
- [6] 范丽,王燕燕. 基于改进蚁群算法的集群负载均衡研究[J]. 计算机时代, 2007, (10): 11-13.
- [7] Han K H, Kim J H. Quantum-inspired Evolutionary Algorithms with a New Termination Criterion[J]. IEEE Transactions on Evolutionary Computation, 2004, 8(2): 156-169.
- [8] 邹谊,魏文龙,李斌,等. 多目标量子编码遗传算法[J]. 电子与信息学报, 2007, 29(11): 2688-2692.

编辑 陆燕菲