维普资讯 http://www.cqvip.com

具有模糊处理时间的网格任务调度免疫算法*)

李 季[1,2] 钟 将[1] 吴中福[1]
(重庆大学计算机学院 重庆 400044)[1]
(重庆教育学院计算机现代教育系 重庆 400067)[2]

摘 要 目前,网格计算作为一种新的计算范式正在兴起。任务调度是其中的一个重要研究领域。该文以AIS的克隆选择算法为基础,给出了基于人工免疫系统的网格任务调度算法。首先,对网格任务调度问题进行模糊化,并给出了形式化描述,随后用结构化的语言对算法进行了说明,最后通过仿真实验对算法的有效性以及算法参数对性能的影响进行了验证。

关键词 人工免疫系统,网格计算,任务调度

An Artificial Immune Algorithm for Job Scheduling in Grid Environment with Fuzzy Processing Time

LI Ji[1,2] ZHONG Jiang[1] WU Zhong-Fu[1]
(Computer College, Chongqing University, Chongqing 400044)[1]
(Department of Computer and Modern Education Technology, Chongqing Education College, Chongqing 400067)[2]

Abstract As a new paradigm, Grid Computing (Computational Grid) is springing up, in which, job scheduling is an important research field. Based on clonal selection algorithm of artificial immune system (AIS), this paper proposed an artificial immune algorithm for resouce and job scheduling in grid environment, which is started with fuzzy and formalized description of grid resource and job scheduling problem, then followed with structured illumination of algorithm. As a result, both of the validity of algorithm and the influence of its parameter on algorithm performance are validated via the simulation studies.

Keywords Artificial immune system, Grid computing, Job scheduling

1 引言

计算网格作为求解科学、工程以及经济学方面的具有巨大挑战性应用问题的一种新的计算范式而正在逐渐兴起[1]。它是满足未来逐渐增加的计算需求的最终框架平台[1~3]。为了满足对计算能力逐渐增加的需求,地理上分布的资源需要逻辑地耦合在一起作为一个整体而工作。随着Internet和WWW持续不断的指数级增长、通信带宽的不断增加、强大计算能力的计算机和低成本组件的广泛应用等,这些有利条件更进一步推动了计算网格成为现实。网格中的计算资源在地理上分布于不同的所有者,每个所有者可以制定自己的资源访问策略、访问费用和各种约束。每个资源所有者有其自己独特的管理和调度资源的方法,网格调度器能够确保它们不与所有者的策略相冲突。在最坏情况下,资源所有者对不同的网格用户收取不同的访问费用,并且随时间的变化而变化。

网格的资源调度问题分为两大类:以性能为目标的调度和以经济为目标的调度。多数的网格系统(尤其是一些研究目的网格)的调度都是第一类的,其任务调度算法将许多影响调度性能的因素忽略掉(例如,网络流量、资源当前负载、资源价格等),而只考虑资源的处理速度和任务的长度这两个关键因素。它们以最小化整体执行时间来匹配任务与资源的映射,由于任务分配和资源调度问题是NP难的组合优化问题[6],当前主要采用启发式方法,包括模拟退火、塔布搜索、遗传算法等等。而基于经济学的调度方法将资源成本作为优化的条件准则。例如,Nimrod/G[4]中的资源代理器允许用户指定自己的预算约束条件、截止时间约束条件等,优化调度依赖于资源使用成本和用户需求规格的折中[5]。但它将任务的执行时间考虑为固定不变和确定性的。

在实际网格环境中,资源价格是影响任务运行性能的一个关键因素。用户不可能为了提高速度,而不顾预算,资源成本是一个必须满足的约束条件。同时,网络流量和资源当前负载都是动态变化的,它们反映出的结果就是影响任务的执行速度,也就是说,任务的执行速度也是动态变化的。因此,网格任务调度也是一个模糊调度问题。

在网格环境下,要确实获取可用计算资源的处理速度以及用户应用任务的处理时间等信息是一项困难的工作。通常,获得可用资源的速度信息还比较容易,但要获取用户任务的计算处理时间就相当复杂。因此,在运行调度任务时,需要从用户的应用规格说明或历史数据中动态地、近似地估计任务长度。

在本文中,作者提出了具有模糊处理时间的人工免疫网格任务调度算法。该算法是受De Castro和Von Zuben所提出的克隆选择算法的启发[7],作者对它作出了修改和扩展并将其应用在网格任务模糊调度中。

*)基金项目:重庆市应用基础研究项目(7969);重庆大学基础及应用基础研究支持项目。李 季 博士生,主要研究方向为网格计算、分布式计算机网络。钟 将 在读博士,主要研究方向:数据挖掘、信息网络安全。吴中福 教授,博士导师。

维普资讯 http://www.cqvip.com

2 网格任务调度问题的定义

类似于 Nimrod-G[8]，我们所研究的网格环境包含多个资源和用户实体，任务的调度是基于市场经济条件的。用户创建和发起包含应用规格说明和服务质量需求（具有优化策略的时间和预算约束）的应用。假定，用户的应用是 task-farming 类型，其包含了 n 个独立的用户任务（jobs）。而每个任务在各个资源上的运行时间受多种因素的影响是动态变化的，根据模糊集理论，运行时间可以用模糊数表示。同时，网格任务的执行要受用户的预算的约束，一般来说，用户预算是确定的，执行一个应用不能超过其最大预算值。

在上述假定情况下，将网格计算环境作如下的形式化描述：需要调度的任务总数为 n，资源总数为 p。其中，$J=\{j_1, j_2, \cdots, j_n\}$ 表示 n 个独立的用户任务，$M=\{m_1, m_2, \cdots, m_p\}$ 代表 p 个计算资源的集合，$\tilde{t}_{ij}$ 表示任务 $i(=1,2,\cdots,n)$ 在资源 $j(=1,2,\ldots,p)$ 上的运行时间。$\tilde{t}_{ij}$ 是三角形模糊数，其成员函数 μ 如图 1 所示。

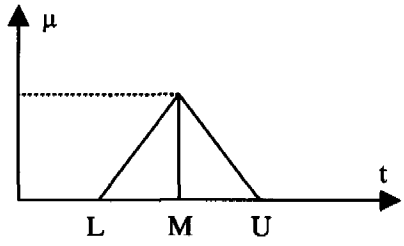

图 1 三角型成员函数

这里，任务的处理时间可以用下面的成员函数来表示：

$$\mu_{i,j}(x)=\begin{cases}0 & x\leqslant L, x\geqslant U\\ \dfrac{x-L}{M-L} & L<x\leqslant M\\ \dfrac{U-x}{U-M} & M<x\leqslant U\end{cases} \tag{1}$$

任务的一种调度策略可以用 $n\times p$ 的 0-1 矩阵 X 表示，其中，p 为资源总数，n 为任务总数，矩阵元素 $x_{i,j}$ 表示任务 j_i 分配到资源 m_j 上进行运行。当且仅当任务 j_i 被分配给计算资源 m_j 时，$x_{i,j}=1$；否则，$x_{i,j}=0$。矩阵的任意一行必有一个元素为 1，也只能有一个元素为 1。

假定 X 代表所有可能的调度方案集 π 中的一种调度策略，则在该调度策略下，网格系统完成全部任务 J 所需的执行时间 C 由下式给定：

$$C(X)=\max_{j\in(1\cdots p)}\left(\sum_{i=1}^{n}x_{i,j}\tilde{t}_{i,j}\right) \tag{2}$$

网格上的每个资源的运行价格是由资源的所有者所决定的，其单位时间的运行价格由 B_j 表示 $(j=1,2,\cdots,p)$。运行一个应用的总的预算由 B 表示。

因此，网格系统的任务调度问题，就是以下式为目标函数的最优化问题：

$$\min_{X\in\pi} C(X) \tag{3}$$

$$\text{s.t.}\ \sum_{j=1}^{p}\sum_{i=1}^{n}x_{i,j}\tilde{t}_{i,j}b_j\leqslant B \tag{4}$$

为了求解该最优化问题，在本文中，采用模糊目标满意度评估函数进行求解。

2.1 模糊目标满意度

对某个调度策略来说，如果应用的完成时间低于 $T_{\min}$，则它就是一个好的调度方案。这是一个模糊目标，其模糊成员函数 μ_G 如图 2 所示。其中，$T_{\min}$ 表示用最快的资源并行处理所有的任务所需的时间；$T_{\max}$ 表示用最慢的资源串行处理所有的任务所需的时间。

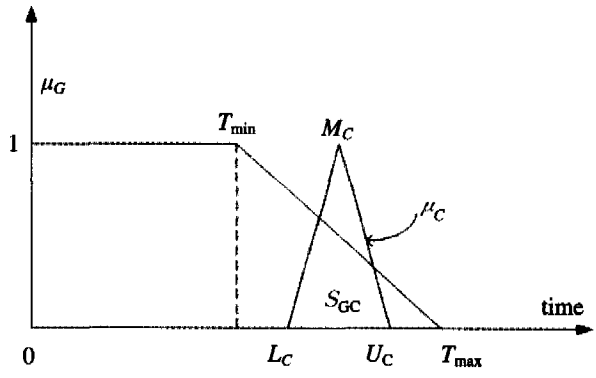

图 2 模糊目标和模糊完成时间之间的重叠区

由于每个任务的处理时间是三角形模糊数，从模糊集理论可知，$C(X)$ 也必然为三角形模糊数，其成员函数 μ_C 如图 2 所示。

模糊目标和模糊处理时间的重叠面积 S_{GC} 如式(5)所表示：

$$S_{GC}=\int_{t_C}^{t}(\mu_G(x)\wedge\mu_C(x))\mathrm{d}x$$

$$t=\begin{cases}T_{\max}, & \text{if } T_{\max}\geqslant U_C\\ U_C, & \text{if } T_{\max}<U_C\end{cases} \tag{5}$$

如果 μ_G 和 μ_C 之间的重叠区域越大，则 S_{GC} 越大。因而，模糊目标满意度 f_A 可以用 S_{GC} 对 S_C 的比率表示：

$$f_A=S_{GC}/S_C, f_A\in[0,1] \tag{6}$$

模糊目标满意度可以作为本文的调度决策评估函数。市场条件下的模糊任务调度问题就是在预算约束的条件下最大化其调度决策评估函数。即：

$$F=\mathrm{MAX}(f_A) \tag{7}$$

求解该类最优化问题的方法，一般分为两类：获得最优解，或获得次优解（速度较快）。前者包括完全搜索、分支限界算法、回溯法等等；而后者包括模拟退火、塔卜搜索以及遗传算法等。

为了有效地求解该调度问题，我们应用免疫原理来进行问题搜索求解。并作如下规定：

每个计算资源的运行速度用单位时间执行的指令数（MIPS）表示，资源的运行价格用每单位时间元表示，应用任务的大小用指令条数（百万）表示。任何任务 j_i 在资源 m_j 上的运行不能中断暂停。下面给出算法的具体说明。

3 免疫调度算法

生物免疫系统是一个高度进化的生物系统，旨在区分外部有害抗原和自我组织，从而清除病原并保持有机体的稳定。从计算的角度来看，生物免疫系统是一个高度并行、分布、自适应和自组织的系统，具有很强的学习、识别、记忆和特征提取能力。将这些特点应用到调度问题中是非常有用的。

本文中提出了一种网格免疫调度算法（Grid Immune Scheduling Algorithms，GISA），来求解网格计算的任务分配问题。算法思想主要遵循两个免疫原理：克隆选择（Clonal Selection，CS）和亲和度成熟（Affinity Maturation，AM）。CS

维普资讯 http://www.cqvip.com

的思想就是最佳抗体被选择进行克隆，克隆的数量与抗体的亲和度(与抗原之间的)成正比；AM 的思想就是新生成的抗体进行超变异(变异概率很高)，变异率同抗体的亲和度成反比，随后那些和抗原之间亲和度较高的变异抗体得到保留。

基于这两个原理可以对组合优化问题进行建模。文[6]提出了最基本的 CS 算法，本文对其进行了改进，将其应用到网格计算任务调度中。算法如下：

网格免疫调度算法 GISA(Grid Immune Scheduling Algorithms)

```
1. 随机生成大小为 K 的抗体种群 P;
2. For 每一代种群 do
{
  2.1  计算每个抗体的亲和度;
  2.2  根据亲和度选定 N 个最佳抗体，构成子种群 PN(在选择最佳抗体时，必须满足预算约束条件);/* N 为算法的参数
  2.3  For each 抗体 in PN do
       {
         2.3.1  根据每个抗体的亲和度克隆复制抗体，放入临时种群 C 中;/* 每个抗体的克隆数量和该抗体亲和度成正比。
       }
  2.4  For each 抗体 in C do
       {
         2.4.1  对种群 C 的每个抗体进行超变异，形成种群 C*;/* 变异概率与亲和度成反比。
       }
  2.5  从种群 C* 中选择亲和度得到改进的抗体，形成免疫记忆细胞群 M;/* 注意预算约束条件的满足
  2.6  将原种群 P 中亲和度最低的|M|+D 个抗体用免疫记忆细胞群 M 和随机产生的 D 个抗体进行替换而构成下一代新的种群 P;/* 引入随机产生的抗体，是为了避免陷入局部最优，其中，D 也是算法的参数。
}
3. Until 算法满足停止条件;
4. 从种群中选择具有最佳亲和度的抗体作为算法的解;
```

对网格计算中的任务调度，可以用固定长度为 N(N=任务数)的整型字符串来生成抗体(原问题的解)的染色体，其中，每个基因位代表任务所分配的计算资源编号。

算法亲和度计算如下：

$$\text{亲和度 } A(k)=f_A(k) \tag{8}$$

其中，$f_A(k)$是抗体 k 所代表的可行解的模糊目标满意度(在一种调度策略下)。

变异算子的选择对算法的收敛速度具有很大的影响。算法可以有多种变异方式，例如，单点随机变异、两点随机变异等等。在这里我们选择单点随机变异，变异概率 $P_m=1-A(k)$。

为了加快算法的收敛速度，在生成初始种群时，引入了一个特殊抗体。该抗体是采取将任务按大小(执行时间)进行排序，最大的任务分配给最快的机器，其余的以此类推这种方式获得的。其余的抗体随机生成。

抗体的克隆数量由下式确定：

$$NC(k)=Index(k) \tag{9}$$

其中，$NC(k)$是抗体 k 的克隆数量；$Index(k)$是抗体 k 在种群中按亲和度进行排序(升序)所得的序号。序号越大，说明了抗体亲和度越大，抗体克隆的数量就越多，这符合免疫学原理。这里我们选择克隆数量按亲和度线性增加。

GISA 算法停止条件可以采用遗传算法的停止条件，我们采用 50 代后停止运行。具有最佳亲和度的抗体即为调度的解。

4 仿真实验

4.1 实验设置

基于上述网格资源免疫调度算法 GISA，我们设计了算法的仿真实验。算法编程用 Matlab 语言实现，实验平台为奔腾Ⅳ 2.4G CPU、256M 内存的微机。首先，为了说明算法的有效性，我们将问题域空间(也就是抗原空间)选定为文[9]中的 DBC 算法所采用的实验数据。也就是 200 个任务在 10 个计算资源上的调度问题。抗体种群规模 $K=50$；选定算法参数 $N=10$，$D=5$。实验结果如图 3 所示。

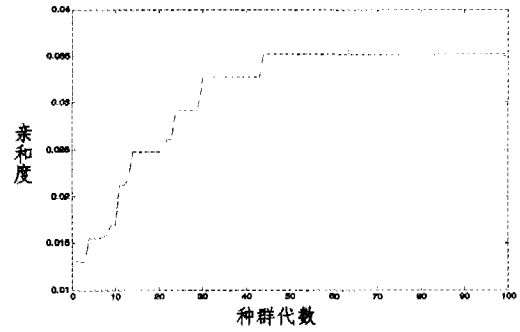

图 3 亲和度/代数图

从图 3 可以看出，亲和度是随着代数的增加而提高的，并且最后趋于平缓。实验证明了算法的有效性。

4.2 参数设置

算法参数 N，D 对算法性能有很大影响，我们分别对 $N=5$，$N=10$，$N=15$(当 $D=5$ 时)和 $D=5$，$D=7$，$D=9$(当 $N=10$ 时)作出了实验对比，如图 4 和图 5 所示。

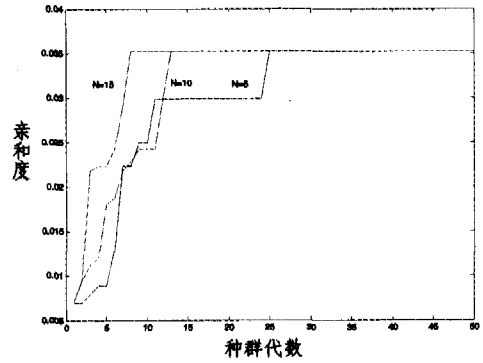

图 4 N 参数对比图

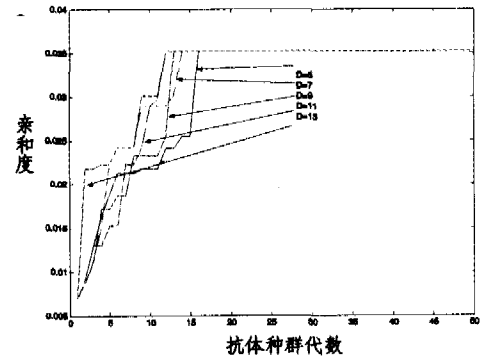

图 5 D 参数对比图

图 4 说明，当 N 增加时，算法的收敛速度也在增加。这是因为，如果选取更多的抗体作为生成记忆细胞的种子，那么将增大获得最优解的概率，这符合免疫学原理。虽然下一代抗体的搜索空间得到加大，有助于减小限于局部最优的概率，但在实验中整个算法的运行时间开销也明显增加。因此，N 参数的确定，应根据问题的特定环境。一般来说，如果对时间要求比较紧迫，N 取整个种群大小的 10%～15%。

(下转第 64 页)

维普资讯 http://www.cqvip.com

其他环境设置不变。让 H2 和 S2 之间建立一个 TCP 流，每个 TCP 连接分别传输 2kB，4kB，6kB，8kB 大小的文件各 20 次，记录文件传输完成的时间。

图 3 是仿真试验的结果。从图中我们可以看到，文件越小，TCP Vages 所花的时间相对越多些，随着文件的增加，二者的差别越来越小。这是因为，文件越大，大多数报文都是最大报文长度，两种算法的差距就很小了。同时对 ftp 协议而言，传输文件差不多是单向数据流，也没有延迟 ACK 干扰了。

结束语 在基于迟延的拥塞避免算法中考虑传输迟延和延迟 ACK 的影响，可以提高预测网络拥塞的准确性。本文经过分析，考虑了不同报文长度以及捎带 ACK 对 RTT 采样的影响，改进了 DCA 的算法，提高了这种算法的性能。仿真试验证明，在网络数据报文长度经常变化的传输流中，本算法表现出很大的性能改进，能够明显提高网络的总吞吐量，增加 IP 网络的资源利用率。

在仿真试验中我们还发现，计算出的等效带宽 B 有时候差别较大，但并不影响本算法的最终优势。因为在网络拥塞时，计算的等效带宽明显偏小，但网络不拥塞的时候，计算出 $T_{q2}-T_{q1}<0$，根据算法，我们需要增加流量，从而保持了高吞吐量。当然，在这个时候我们需要重新估计等效带宽的大小，以便计算的拥塞迟延偏差更加准确。

参 考 文 献

1 Jacobson V, Karels M J. Congestion avoidance and control. IEEE/ACM Trans on Networking, 1988, 6(3): 314 ~ 329

2 Stevens W. TCP slow start, congestion avoidance, fast retransmit, and fast recovery algorithms. Internet Network Working Group, RFC 2001, 1997

3 Xu W, Qureshi A G, Sarkies K W. Novel TCP congestion control scheme and its performance evaluation Communications. IEE Proceedings, Aug. 2002, 149(4): 217～222

4 Leung Fei Peng. A novel fair bandwidth allocation algorithm for TCP window control V. C. M. In: Performance, Computing, and Communications Conference, 2003. Conference Proceedings of the 2003 IEEE International, April, 2003. 317～324

5 Wu Cheng-Shong, Hsu Ming-Hsien, Chen Kim-Joan. Traffic shaping for TCP networks: TCP leaky bucket. TENCON '02. Proceedings. 2002 IEEE Region 10 Conference on Computers, Communications, Control and Power Engineering. 2002, 2(28-31): 809 ～ 812

6 Chan A C F, Tsang D H K, Gupta S. TCP (transmission control protocol) over wireless links. Vehicular Technology Conference, IEEE 47th On 1997, 3: 1326～1330

7 Srivastava A, Friday R J, Ritter M W, Filippo W S. A study of TCP performance over wireless data networks. Vehicular Technology Conference, 2001. VTC 2001 Spring. IEEE VTS 53rd, 2001, 3(6-9): 2265～2269

8 李云，陈前斌，隆克平，等. 一种基于链路带宽估计的 TCP 慢启动算法. 计算机学报，2003，26(6)：693～700

9 陈晶，郑明春，孟强. 一种基于历史连接的网络拥塞控制算法及其性能分析. 计算机研究与发展，2003，40(10)：1470～1475

10 Brakmo L S, Peterson L L. TCP Vegas: end to end congestion avoidance on a global Internet. Selected Areas in Communications, IEEE Journal on, 1995, 13(8): 1465～1480

11 Wang Z, Crowcroft J. Eliminating periodic pACKet losses in the 4.3-Tahoe BSD TCP congestion control algorithm. Comput. Commun. Rev., 1992, 22(2): 9～16

12 Martin Jim, Nilsson Arne a, Rhee Injong. Delay-Based Congestion Avoidance for TCP. IEEE/ACM Transactions On Networking, 2003, 11(3): 356～369

13 paxson V. Measurements and analysis of end-to-end Internet dynamics: [Ph. D dissertation]. UC Berkeley, 1996

(上接第 37 页)

当 D 变大时，引入下一代的随机抗体数量得到增加，这主要是为了增加多样性，它可以有助算法陷入局部最优。但从图 5 看，参数 D 对算法的收敛速度并没有明显的影响，并且算法的执行时间也没有什么明显的变化。

4.3 与 DBC 算法的性能比较

为了考察 GISA 算法的性能，我们选择 DBC 算法为比较对象。二者的任务数和计算资源数均分别为 200 和 10，总预算为 6000。表 1 比较了两种算法的性能，数据为 20 次仿真实验的平均值。

表 1 2 种算法的性能比较

	DBC 算法	GISA 算法
平均任务完成时间	2727	2148
算法本身执行时间	19	186

可以看出，GISA 算法由于引入免疫原理，使得任务完成时间有了明显的提高。但算法本身的运行时间却比 DEC 算法高出一个数量级，这是算法本身的搜索特性决定的。对于大规模的科学计算问题，这点时间上的开销，却可以带来系统整体运行效率的极大改善。

结论 本文针对网格资源管理系统的任务分配调度问题进行研究，提出了 GISA 算法。其核心思想是用人工免疫系统求解任务分配问题。作为一种智能优化搜索策略，人工免疫系统(AIS)在函数优化、组合优化、调度问题等方面得到了广泛应用并取得了很好的效果。在大多数的情况下，免疫算法取得了比现有启发式算法更好的求解结果[9]。

GISA 算法考虑到了网格任务调度问题中存在的很多影响任务运行速度的因素，例如：网络线路带宽、资源的负载等等，而将任务的运行时间作为模糊时间。因此，算法就是对具有模糊运行时间的任务进行分配调度，并在用户预算约束条件下，通过免疫原理来求解该 NP 完全问题。算法从仿真实验来看，是可行和有效的，并且收敛速度较快(一般来说，比遗传算法较快)。理想中的网格应该还具有资源预定功能。因此，我们下一步的研究工作就是将资源预定功能结合到算法中，以期得到较理想的调度算法。

参 考 文 献

1 Foster I, Kesselman C. The Grid: Blueprint for a New Computing Infrastructure. Morgan Kaufmann, 1998

2 Buyya R, Abramson D, Giddy J. Grid Resource Management, Scheduling, and Computational Economy. International Workshop on Global and Cluster Computing, Japan, 2000

3 Baker M, Buyya R, Laforenza D, The Grid: International. Efforts in Global Computing. Intl. Conf. on Advances in Infrastructure for Electronic Business, Science, and Education on the Internet, Rome, Italy, 2000

4 Abramson D, Buyya R, Giddy J. A Computational Economy for Grid Computing and its Implementation in the Nimrod-G Resource Broker. Future Generation Computer Systems Journal, Volume Elsevier Science, 2002, 18(8): 1061～1074

5 Buyya R, Abramson D, Giddy J. An Economy Driven Resource Management Architecture for Global Computational Power Grids. Intl. Conf. on Parallel and Distributed Processing Techniques and Applications (PDPTA'2000), Las Vegas, USA, 2000

6 Ibarra O H, Kim C E. Heuristic algorithms for scheduling independent tasks on non-identical processors. Journal of the ACM, 1997, 24(2): 280～289

7 De Castro L N, Von Zuben F J. Clonal selection algorithm with engineering applications. GECCO 2000 Workshop proceedings, 2000. 36～37

8 Buyya R, Abramson D. An Economy Driven Resource Management Architecture for Global Computational Power Grids. Intl. Conf. on Parallel and Distributed Processing Techniques and Applications, Las Vegas, USA, 2000

9 Buyya R, Murshed M. A Deadline and Budget Constrained Cost-Time Optimize Algorithm for Scheduling Parameter Sweep Applications on the Grid. GridSim Toolkit Release Document, Dec. 2001

10 肖人彬 王磊. 人工免疫系统：原理、模型、分析及展望. 计算机学报，2002，25(12)：1281～1293