

# 基于免疫 Agent 的网格性能抗衰

徐建,张琨,刘凤玉

(南京理工大学计算机科学与技术学院,南京 210094)

[摘要] 模拟生物免疫机理实现计算网格的性能抗衰是一个崭新的研究方向。分析和比较了免疫和抗衰之间的异同,构建了基于免疫 Agent 的系统抗衰逻辑模型,模拟生物免疫机理对计算网格的资源和性能进行监控和诊断,建立性能衰退的数学模型,并给出了抗衰策略决策方法。以一个分布式的音像资源事务处理系统为背景进行应用研究,给出了一个两阶段超指数分布的数学模型来评估性能,结果表明该方法是有效的。

[关键词] 网格;免疫系统;性能监控;软件抗衰;免疫 Agent

[中图分类号] TP 302.7 [文献标识码] A [文章编号] 1009-1742(2007)11-0087-05

## 1 引言

计算系统性能衰退现象是指一个长时间持续运行的计算系统会发生状态退化和性能降低,最终导致系统崩溃。发生衰退现象的主要原因之一是系统资源的耗尽<sup>[1]</sup>。网格作为一种分布式高性能计算环境,应该具备的两个重要特点是协调资源使用以及提供非平凡质量的服务。为此开展网格环境下系统性能抗衰或自愈研究。研究网格计算环境下性能抗衰的一般途径是实时地监控网格资源使用状况和节点性能,建立网格性能衰退的模型,进行合理的资源调度,预测最优的执行自愈操作的时间。

文献[2]在网格监控体系结构的基础上,提出了两层资源监测系统,使用有向无环图描述资源间的依赖关系并作为系统的逻辑基础,满足了网格监测的需求,以较低的系统开销获得较多的功能,具有较强的灵活性和扩展性。文献[3]分析了目前主要的网格监测系统,阐述了网格监测的各种需求及其相互关系,基于 R-Net 实验平台,设计了网格监测系统 RNMS,使用诸如自适应传感器控制、按需测量、分布式目录服务、节点组管理和混合数据格式等技术,在提高扩展性和互操作性的同时降低了侵扰。

文献[4]提出了一种基于 Agent 的计算网格资源管理方案,该方案实现了网格资源管理的 3 个网格协议原型——服务登记协议,服务发现协议和服务访问协议,为网格的计算资源和服务提供一个统一的高层管理框架,为网格资源管理提供了一种有效的方法和途径。文献[5]应用一种不依赖于数学模型的模糊控制技术,对很难建模的复杂对象,利用人的经验知识来完成调节任务,并据此给出了一个实现原型,具有很好的可扩展性和移植性。

笔者吸收了生物免疫系统的免疫机制在资源监控和节点性能感知的基础上,建立数学模型来刻画系统性能衰退,并给出执行抗衰策略的最优时间估计方法,提高了系统的可用性,并降低了维护成本。

## 2 免疫和抗衰

免疫是机体识别“自我”与“非我”抗原的一种生理功能,对自身抗原形成天然免疫耐受,对“非我”抗原产生排异作用。免疫机制是免疫系统在识别和清除“非我”抗原过程,产生的各种生物学作用的总称,免疫是通过应答来完成的。

计算网格是一个由成百上千个计算节点、资源组成的复杂的分布式系统,实现计算功能的各个功

[收稿日期] 2006-06-12; 修回日期 2006-10-17

[基金项目] 国家自然科学基金资助项目(60273035), 国防科工委基础应用项目(K1704060511)

[作者简介] 徐建(1979-),男,江苏江阴市人,南京理工大学计算机学院博士生,主要研究方向是软件抗衰,信息安全

能模块分布在整个网络。长期运行会由于资源泄漏、程序 bug、不适当的维护性操作导致计算系统性能衰退。在具备抗衰机制的计算网格中,存在一个感知系统性能,识别性能衰退,建立性能模型,实施抗衰操作的过程,这一过程与生物体的免疫过程存在着很多的相似之处,可概括如表 1 所示。

表 1 免疫机理与系统抗衰的类比

Table 1 The analogy of immune mechanism and system rejuvenation

生物体免疫	计算网格抗衰
生物体	计算网格
细胞	计算节点
抗原	性能衰退
抗体	抗衰策略
抗原识别	性能衰退检测
肽链	性能数据
记忆性	衰退模型
免疫应答	抗衰策略的实施

从表 1 类比可见,免疫系统具有的分布性、并行性、应答性和记忆性等特点完全符合计算网格抗衰的需求。另外,从计算的角度看,基于免疫机理构建的抗衰系统可看作一个并行的、分布式的处理系统,实现了对复杂计算系统的持续实时的控制。对免疫机制的研究主要是理解免疫系统对抗原的识别过程、学习和记忆机制,并在抗衰过程中进行模拟和应用。

### 3 基于免疫 Agent 的抗衰模型

计算网格抗衰的目的在于提高计算系统的可靠性、可用性。其关键是：**a.** 如何获取资源使用信息和节点性能信息；**b.** 如何建立性能衰退模型；**c.** 何时采用何种粒度的抗衰策略。笔者给出一个基于免疫 Agent 的抗衰模型来实现系统抗衰,其逻辑结构如图 1 所示。

实现抗衰的免疫 Agent 由 5 个软件 Agent 组成,其中感知 Agent 模拟生物体细胞自我感知功能,监控计算网格资源使用和计算节点性能;识别 Agent 借鉴阴性选择机制<sup>[6]</sup>模拟免疫识别过程,验证性能衰退出现;记忆 Agent 模拟生物体的联想记忆功能建立系统衰退的数学模型;选择 Agent 模拟免疫的不同层次选择合适抗衰粒度的抗衰策略;应答 Agent 模拟免疫应答过程执行抗衰操作。一个完整的抗衰过程是由性能感知、衰退检测、衰退模型刻画、抗衰决策和抗衰实施这五个子过程构成的,基本遵循了免疫过程,唯一不同之处在于抗体知识库的构建,在免疫系统中抗体知识库是不断更新的,而在抗衰系统中对应的抗体是抗衰策略,系统预先产生了若干

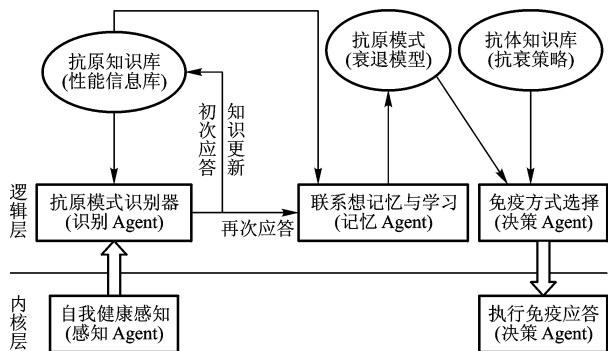


图 1 免疫 Agent 的抗衰逻辑模型

Fig. 1 The logical model based on immune agent for rejuvenation

不同粒度的静态抗衰策略。因此,总的说来,基于免疫 Agent 的抗衰模型遵循了生物免疫机理,具有一定的合理性和可行性。记忆 Agent、决策 Agent 的实现原理将是阐述的重点,而对感知 Agent、识别 Agent 以及应答 Agent 只给出简要的模型和算法。

感知 Agent 在实现上由传感器、数据封装器、事件处理器、守护进程、过滤器和通信机制组成,如图 2 所示。

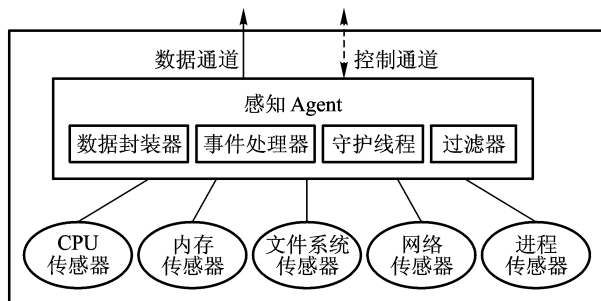


图 2 感知 Agent 的逻辑模型

Fig. 2 The logical model of self-aware Agent

图 2 中,传感器负责收集资源使用情况和计算节点性能数据;数据封装器中把采集的数据转为二进制的位串(入侵抗原);事件处理器用来协调多个 Agent 之间的交互;通信机制则用来发送任务,传递信息。感知 Agent 有两个功能：**a.** 收集系统资源的消耗情况(可用内存、页面缓冲池、非页面缓冲池、句柄、逻辑磁盘可用空间等)等资源的使用情况和负载数据；**b.** 为识别 Agent 提供大量的历史信息数据。

生物体免疫功能是通过阴性选择机制来实现“自我”和“非我”的免疫识别。识别 Agent 的作用就是借鉴这种阴性选择机制模拟免疫识别过程,把

健康的系统性能状态作为“自我”,把处于衰退过程的性能状态作为“非我”,应用基于免疫识别的检测方法实现计算系统性能衰退检测。识别 Agent 实现基于 NIS 的性能异常检测<sup>[7]</sup>。应答 Agent 模拟了生物体消灭外来抗原的过程,关键之处在于检测点技术,即保存实施抗衰前用户请求的状态,在抗衰后能重新启动对这些请求的响应处理。

### 3.1 记忆 Agent

记忆 Agent 模拟生物体的联想记忆功能,协助识别 Agent,建立衰退数学模型来评估性能状态。识别 Agent 的功能只能验证计算系统的衰退是否出现,同时把衰退过程粗略地划分成不同的状态,缺少不同衰退状态之间的联系。记忆 Agent 采用基于时间和负载的方法建立衰退的数学模型,该方法可以通过以下 3 个步骤来实现。

Step 1 使用 5 元组 (AvgLoad, CPUctxSwitch, NetTraffic, PageSwitch, IONum) 来度量系统在特定的监控时间段内的负载,其中各个变元的含义如下:

- AvgLoad 特定的监控时间间隔内平均的 CPU 负荷;
- CPUctxSwitch 特定的监控时间间隔内 CPU 上下文切换的次数;
- NetTraffic 特定的监控时间间隔内的网络流量,包括接收的数据包数目和发送的数据包数目;
- PageSwitch 特定的监控时间间隔内发生内存缺页时,物理内存和交换空间之间页面换入数目和换出数目的总和;
- IONum 特定的监控时间间隔内读写磁盘的次数。

Step 2 应用  $k$  平均聚类算法对负载序列进行聚类,聚类后得到的不同簇作为不同的衰退状态  $SA_i$  ( $1 \leq i \leq S$ ),簇数目  $S$  即为衰退状态空间的大小,其中衰退状态也包括系统启动时的健康状态;

Step 3 计算在衰退状态  $SA_i$  下不同时刻的驻留时间,并拟合得到不同状态下的驻留时间的分布函数,从而建立衰退状态之间的数学模型。

通过 Step 1 和 Step 2 得到了状态有限的状态空间,并把每个负载样本都划分到特定的衰退状态,则每个样本所属的状态就构成一个随机过程  $\{X(t), t \geq 0\}$ ,该随机过程从给定的状态转移到另一个特定状态的概率仅依赖于当前的状态,而与如何到达那个特定状态无关,因此这是一个马尔可夫过程。又因为聚类算法得到的状态空间是离散的,并且状态

之间的转移能发生在任意时刻,与时间无关,所以这个过程是一个齐次的连续参数马尔可夫链。对于齐次的连续参数马尔可夫链其驻留时间服从指数分布,如果取消这一限制,则允许任意分布,笔者就采用两阶段的超指数分布函数来拟合驻留时间,其密度函数为

$$f(t) = \sum_{i=1}^2 p_i \mu_i e^{-\mu_i t}, \quad t > 0 \quad (1)$$

其期望值由

$$E(X) = p_1/\mu_1 + p_2/\mu_2 \quad (2)$$

来计算,方差系数由

$$C_x = \sigma(X)/E(X) \quad (3)$$

计算,其中  $\sigma(X)$  为标准差,最后得到的驻留时间分布函数为

$$F(t) = 1 - p_1 \exp[-\mu_1 t] - p_2 \exp[-\mu_2 t] \quad (4)$$

由式(1)至式(3)可计算得到系数

$$p_1 = (1 - ((c_x^2 - 1)/(c_x^2 + 1))^{1/2})/2,$$

则  $p_2 = 1 - p_1, \mu_1 = 2 p_1 (1/E(X)), \mu_2 = 2 p_2 (1/E(X))$ 。

### 3.2 决策 Agent

决策 Agent 负责对何时以及采用何种粒度的抗衰策略做出决策。在生物免疫系统中,免疫包括物理免疫、生理免疫、先天免疫和适应性免疫几个层次,相应的在计算网格中,抗衰策略的粒度通常可划分为 3 类——系统级 Level 1、应用程序级 Level 2 以及构件级 Level 3。三者之间的共同点是都需要在实施抗衰策略之前对必要的用户请求信息加以保存,设置检查点;其差别在于:系统级的抗衰策略是通过重启操作系统来实现,它清除了所有内部状态,彻底的恢复系统的健康,然而抗衰成本是三者中最大的;应用程序级和构件级的抗衰策略在实施后并不能令系统性能恢复到初始的健康状态,而是处于一种亚健康状态,相应的其抗衰成本也较低。选择合适的抗衰粒度对于系统的可用性有很大的影响,因此在抗衰策略选择时需要考虑以下几个因素:**a.** 当前系统性能;**b.** 执行抗衰策略后的系统性能;**c.** 下一时刻的系统负载。另外,还需要考虑的一个问题:不管最终在何时采用何种抗衰粒度,抗衰本身必定也会带来一定的停机时间和成本。因此,提出了基于成本和恢复性能的抗衰策略决策方法。执行抗衰策略的时间可由

$$T = \sum_{i=1}^s \int_0^{\delta} F_i(t) dt \quad (5)$$

计算,表示在进入失效状态前在各个状态上驻留时间的总和, $\delta$  为进入失效状态的时刻。抗衰粒度的

选择可由

$$L = w_1(c_i p(t) + c_0 q(t)) + w_2 h_i \quad (6)$$

来决定,其中  $p(t)$  表示  $t$  时刻系统进入失效状态的概率, $q(t)$  表示  $t$  时刻系统意外失效的概率, $c_i$  ( $i=1,2,3$ ) 表示 3 种不同的抗衰下的恢复成本; $c_0$  表示系统意外失效的恢复成本; $h_i$  表示实施 3 种不同抗衰策略后的健康程度; $w_1, w_2$  分别表示成本和恢复性能的权重。权重的调整依赖于下一时刻的负载,若 Level 2 或 Level 3 级别的策略恢复后的性能不足以为下一时刻的用户请求提供高质量的服务时,就需要实施 Level 1 级的抗衰,此时  $w_2$  就大,否则根据  $L$  的值决定采取 Level 2 或 Level 3

#### 4 应用案例

ATMVS 是一个管理多台 JVC 的音像资源事务处理系统,主要提供音像资源检索和视频点播功能,在长时间的运行后会系统性能衰退,请求的响应时间过长等症状,实验是利用采集的系统运行时度量数据来评价它的性能状态,给出性能衰退的两阶段超指数分布的数学模型。

具体的实现方法:首先,通过感知 Agent 采集上述的负载 5 元组中涉及的系统运行时度量、以及资源耗费数据;然后,应用识别 Agent 进行衰退检测,接着通过记忆 Agent 建立计算系统性能衰退的数学模型,最后决策 Agent 给出预测时间和抗衰策略。图 3 显示的是由感知 Agent 采集的物理内存使用的时序数据,采样间隔是 5 min,总的采集时间为 7 d,在这期间系统没有发生宕机,其他的性能数据不再一一枚举了。

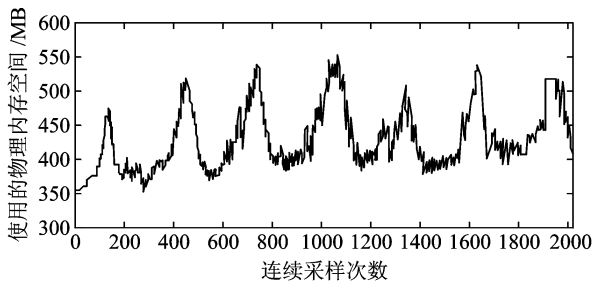


图 3 感知 Agent 采集的物理内存使用的时序数据 (采样间隔为 5 min)

Fig. 3 The time series of used physical memory collected by self-aware Agent every 5 min

对负载 5 元组构成的序列应用聚类算法后得到 9 个不同的簇,如表 2 所示,每一个簇代表不同的衰

退状态,其中初始健康状态也看作一个衰退状态,每个簇包含的样本数目即簇大小表示系统处于该稳定状态下的概率。对于不同的衰退状态,使用超指数分布进行拟合,可以得到驻留时间的分布函数如表 3 所示。

表 2 聚类后得到的不同的衰退状态

簇号	负载变量					簇大小
	Avg Load	CPU Ctx Swtich	Net Traffic	Page Swtich	IO Num	
1	0.14	48 903.4	70.6	0.0	10.3	3.3
2	0.21	54 780.3	65.4	11.2	99.2	9.4
3	0.26	66 311.4	87.4	9.8	99.2	13.4
4	0.25	65 871.0	86.9	10.0	104.3	12.6
5	0.29	66 885.0	98.4	13.2	120.3	45.6
6	0.47	86 235.8	100.3	21.8	160.3	5.3
7	0.49	72 458.9	140.5	33.7	119.9	5.5
8	0.54	71 112.2	160.9	43.8	145.5	3.6
9	0.77	110 783.5	230.7	87.3	181.1	1.3

表 3 不同衰退状态下驻留时间的分布函数

衰退状态	驻留时间的分布函数 $F(t)$
SA <sub>1</sub>	$1 - 0.999 5e^{-0.445 702 3t} - 0.000 5e^{-0.000 222 9t}$
SA <sub>2</sub>	$1 - 0.998 7e^{-0.327 565 6t} - 0.002 3e^{-0.000 754 3t}$
SA <sub>3</sub>	$1 - 0.841 3e^{-0.327 497 6t} - 0.158 7e^{-0.061 646 6t}$
SA <sub>4</sub>	$1 - 0.873 2e^{-0.576 346 3t} - 0.126 8e^{-0.083 692 9t}$
SA <sub>5</sub>	$1 - 0.978 3e^{-0.723 621 1t} - 0.021 7e^{-0.016 048 5t}$
SA <sub>6</sub>	$1 - 0.794 5e^{-0.632 222 1t} - 0.205 5e^{-0.163 526 2t}$
SA <sub>7</sub>	$1 - 0.863 7e^{-0.578 802t} - 0.136 3e^{-0.913 400 0t}$
SA <sub>8</sub>	$1 - 0.821 1e^{-0.834 345 5t} - 0.178 9e^{-0.181 786 1t}$
SA <sub>9</sub>	$1 - 0.883 4e^{-0.611 002 3t} - 0.116 6e^{-0.080 646 2t}$

不同衰退状态下拟合得到的驻留时间分布函数即衰退的数学模型,该模型可以用来预测下一时刻所处的状态,计算出各个状态下的资源耗费的趋势和斜率,从而能够用来预测资源耗尽的时间。在线处理时,当接收到新的样本数据,首先由识别 Agent 进行衰退检测,时间复杂度为  $O(n)$ ,其中  $n$  为检测字的数目;如果验证到衰退出现,接着通过计算与聚类后中心点的距离确定衰退状态,而后根据该状态下的驻留时间分布函数估计耗费趋势,时间复杂度也为  $O(n)$ ,其中  $n$  为该衰退状态下样本点的数目。将基于模型的抗衰策略实施时间的预测结果与文献

[1]中提出了的基于时间的线性模型预测的结果进行比较,使用由感知 Agent 采集的同一组数据,前者得出经过 7643.9 个 5 min 后实施抗衰策略的结论,而线性模型得出的结论为 9907.6 个 5 min。提出的两阶段超指数分布的模型预测的时间比线性模型预测的耗竭时间要短,这是因为线性模型是基于时间的,完全没有考虑负载对系统资源分配和释放时间的影响。而基于模型的抗衰策略同时考虑了时间和负载对系统的影响,预测的效果更加准确。

## 5 结语

从生物免疫学的角度,吸收了生物免疫的灵感,模拟免疫系统的感知、识别、记忆和应答的机理,实现了计算网格资源监控、衰退检测、性能评估,抗衰实施的全过程,为系统抗衰问题的研究提供了一个崭新的途径。给出了一个基于免疫 Agent 的抗衰逻辑模型,在应用案例中对该模型进行了评价,并与其他模型进行了比较,结果表明新方法是有效且可行的。对于应答 Agent 仅给出了需要解决的关键技术,必须进一步开展这项研究工作,并进一步拓展免疫机理在网格资源分配、性能调整、体系结构方面的应用。

## 参考文献

- [1] Garg S, Moorsel A V. A methodology for detection and estimation of software aging [A]. In Proc of 9th Intl Symposium on Software Reliability Engineering [C]. Paderborn, Germany, November, 1998. 282 ~ 292
- [2] Liu Donghu, Xu Zhiwei, Wei. A dag - based two - layer grid monitoring system [J]. Journal of Computer Research and Development, 2002, 39(8): 937 ~ 942
- [3] Yao Yuanzhe, Du Cuilan, Liu Xinran, et al. R-net grid monitoring and alarming system design [J]. Journal on Communications. 2006, 27(2): 168 ~ 177
- [4] Li Chun, Lu Zhengding, Li Layuan. An agent-based approach for grid resource management [J]. Journal of Wuhan University of Technology (Transportation Science & Engineering), 2003, 27(1): 7 ~ 10
- [5] Tao Muli, Wu Chanle, Xing Jianbing. Adaptive tuning mechanism for grid application based on performance monitoring [J]. Journal of Computer Research and Development. 2004, 41(12): 2175 ~ 2180
- [6] Forrest S. Self - nonself discrimination in a computer [A]. Proceeding of IEEE Symposium on Research in Security and Privacy [C]. Los Alamos, CA: IEEE Computer Society Press, 1994, 202 ~ 212
- [7] 徐建,游静,陈昊,等.基于NIS的异常检测算法[J]. 计算机科学, 2005, 32(4): 140 ~ 143

# An Immune Agent-based Performance Rejuvenation for Grid

Xu Jian, Zhang Kun, Liu Fengyu

( School of Computer Science and Technology, Nanjing University of Science and Technology, Nanjing 210094, China)

[Abstract] Simulating biological immune mechanism to implement performance rejuvenation is a new research approach for computing grid. This paper analyzes and compares the immune process's characteristics and computing system's rejuvenation, sets up the logical model of system rejuvenation based on immune agent above simulates the biological immune mechanisms to monitor resource usage and diagnose performance degradation, constructs the mathematics model, then puts forward a method to make decision for rejuvenation policy. On the basis of above an application regarding audio-video resource transaction process system as background is studied, a two-stage hyper-exponential model is proposed to evaluate system performance. The results of case study indicate that this method is effective and feasible.

[Key words] grid; immune system; performance monitoring; software rejuvenation; immune agent