基于服务级别协议的云资源分配

冯国富 唐明伟 刘林源 韩冰青
(南京审计学院信息科学学院 南京 210029)

摘 要 与网格、集群等传统计算模式不同,云计算为用户提供了一种利用远程计算资源的实用商业模型。在不同的客户之间动态分配云资源池以获得最大收入,成为云服务提供商最为关心的问题。云计算中心需要把面向客户的服务层指标转换为面向系统的操作层指标,根据服务级别协议动态管理云计算资源。研究了基于服务级别协议的服务提供商收入最大化问题,借助排队论模型对资源分配问题进行了形式化描述,然后依据定价机制、服务请求到达率、服务率、可用资源等因素给出了资源最优分配方案。实验结果表明,该算法优于相关算法。

关键词 云计算,服务级别协议,资源分配,定价模型
中图法分类号 TP393 **文献标识码** A

Service Level Agreement Based Allocation of Cloud Resources

FENG Guo-fu TANG Ming-wei LIU Lin-yuan HAN Bing-qing
(School of Information Science, Nanjing Audit University, Nanjing 210029, China)

Abstract Cloud computing is different from the traditional computing models such as grid computing and cluster computing, and provides a practical business model for customers to use remote resource. It is natural for cloud providers to maximize their revenue through allocating the pooled computing resources dynamically. It is required to transform the customer-oriented service level metrics into system-oriented operating level metrics, and control the cloud resources adaptively based on Service Level Agreement (SLA). This paper addressed the problem of maximizing the provider's revenue through SLA-based dynamic resource allocation among the differentiated customers. We formalized the resource allocation problem with queuing theory and proposed the solutions, in which many factors such as pricing mechanisms, arrival rates, service rates and available resources are considered. The experimental results show that our algorithms outperform related work.

Keywords Cloud computing, Service level agreement, Resource allocation, Pricing model

1 引言

云服务提供商通过出租云服务获得收入,客户可通过租用云服务满足应用需求。在云环境下,客户只需租用所需要的服务类型和资源数量。以服务级别协议(Service Level Agreement,SLA)为基础的实用商业模型在云计算模式中占据重要地位,这也是云模式区别于传统网格、集群计算的关键因素[1]。

SLA提供了一种表达客户服务需求指标和服务质量指标的共同平台。SLA通常针对可用性、响应时间等指标在服务提供商和客户之间建立关于服务质量、质量担保、违约责任等问题的一致理解。基于服务质量的定价机制和违约之后的惩罚机制需要在SLA里面明确定义。

服务提供商通常以多租户模型管理租赁服务,物理资源池在用户之间共享,为每个用户分配一台虚拟机,资源池资源在各个用户之间动态分配或再分配。其中用户请求特点、实时可用资源以及与每个用户的SLA约定,是云资源调度管理的依据。

本文定义服务实例为某一类客户请求与所分配虚拟机资源的组合。一个用户可以申请多类服务,所以一个用户可以属于多个服务实例。

每个服务实例都具有不同的属性,例如到达率、执行时间、价格机制,等等。即便对于同一服务实例,其到达率也会随时间动态变化。云服务提供商面临的一个基础且重要的问题是,如何基于性能指标和SLA价格机制,在服务实例之间动态分配物理资源以使得收入最大化。

本文研究云计算中心基于SLA的收入最大化问题。基本思想是依据实时测量的性能指标,在不同的服务实例之间自适应地分配资源池资源。资源分配策略以SLA中的服务层指标价格机制为中介,建立各种操作层指标到服务收入的函数关系,进而控制服务最大化时的物理参数。本文的创新工作主要包括以下两个方面。

到稿日期:2013-06-16 返修日期:2013-10-18 本文受国家自然科学基金(60803111, 61073208),江苏省自然科学基金(BK2013830, BK2011692)资助。

冯国富(1977—),男,博士,副教授,主要研究方向为Internet环境下的资源共享,E-mail:njufgf@gmail.com;唐明伟(1983—),男,博士,讲师,主要研究方向为云计算;刘林源(1981—),男,博士,讲师,主要研究方向为服务组合;韩冰青(1979—),男,博士,副教授,主要研究方向为无线网络与分布式计算。

(1)基于排队论对资源分配问题进行了数学化描述。该形式化模型考虑了应用的资源数量、请求到达率、服务时间和价格模型等因素。

(2)给出了不同服务实例之间基于 SLA 的资源分配最优化算法。

2 相关工作

实用商业模型是云计算区别于传统计算的关键因素[1]，而 SLA 是促成实用商业模式的关键所在。SLA 提供了一定的机制和工具，使得服务提供者和终端用户能够以共同理解的语言表述服务性能需求和基于性能的价格机制。基于商业协议，通过资源动态管理调度，实现云服务收入最大化，自然成为服务提供商关心的问题。

关于商业数据中心的资源管理，目前已有广泛研究。效用(Utility)常用作资源分配的标准，也作为多个系统目标时的折中基准依据。Walsh 等人[6]讨论了数据中心虚拟资源池的动态分配问题。Rajkumar 等人[7]提出了一种多 QoS 应用约束条件时效用最大化资源分配模型 Q-RAM，Ghosh 等人[8,9]改进了 Q-RAM 的性能。

Menascé 等[10,11]提出了一种基于爬山(hill climbing)技术的资源分配算法，可在计算任务波动时，通过系统参数配置使既有资源更好地满足 QoS 需求。Chandra 等人[12]提出基于测量负载和 QoS 要求，在应用之间动态调整服务资源。Levy 等人[13]提出了一种系统性能管理结构和原型，其功用以商业价值服务级别协议为基础，系统动态分配服务器资源，根据性能需求平衡多类负载。Li 等人[14]提出了满足响应时间要求和服务功用要求的自动计算策略，用以降低资源消耗。但这些工作都没有考虑 SLA 中的经济因素。

Zhang 和 Ardagna[15]提出了一种面向自动计算的数据中心资源分配控制器，目标是使基于 SLA 的收入最大化。Liu 等人[16]提出了一种多 SLA 级别服务的收入最大化模型。Time 等人[17]提出了一种计算资源管理框架，结合动态价格机制、任务优先级、客户分类等因素，强化资源管理，提高资源收入。但以上工作均采用了离散价格计费方式，而本文则采用基于服务质量的连续价格计费模式。

有少数相关工作与本文拥有相同的应用场景。Zhu 等人[3]提出了一种在客户之间分配服务器资源的策略，目的是减少平均响应时间。但该工作没有考虑经济因素，而且最优化方案中的权重因素缺少具体物理含义。Villela 等人[18]研究了服务提供者在 QoS 约束条件下，服务器资源在应用之间的调度分配问题，用 M/G/1/PS 队列模拟应用服务器，推导出 3 种增加收入的资源分配方法，但 3 种算法均为近似算法。

文献[4]与本文工作非常接近，提出了两种资源分配策略，即启发式和贪婪算法。贪婪算法最优但是计算复杂度太高而不可用，改进算法效果并非一直最优。Heuristic 算法简单，但在本文工作中其有效性受环境参数因素影响明显。

3 模型假设

云计算应该集成自动化资源管理模型，根据所签约的 SLA 对服务进行动态管理。本文考虑了一种应用场景：云服务提供商以不同 SLA 提供多种服务。每种服务在数据中心以虚拟机形式组织，所耗资源可动态伸缩。目标是在价格机制确定时，在服务实例之间分配以资源池形式存在的计算资源以使得整体收益最大。

3.1 系统模型

假设云计算中心由 N 台相同类型的普通计算机组成。计算机动态组合为集群。一台计算机在某一时刻只能加入一个集群。集群对用户虚拟成为单一服务器，服务于某一特定服务实例。假设每个集群计算能力与计算机数量成正比。这一假设具有合理性，尤其对于那些可拆分并行处理的计算任务。例如，由一些独立运算部分构成的动态网页；或者一些由可拆分运算单元组成的计算任务。

服务提供商与 m 个用户签约长期 SLA。每个服务实例依次分配 $n_1, n_2, \cdots, n_m$ 台计算机。用户无需知道虚拟机的具体操作细节。每个服务实例的服务请求符合泊松分布，到达率为 λ。每次请求服务时间服从负指数分布，单台计算机的平均服务率为 $1/\mu$(其中 μ 为单位时间内处理的请求数量)，则由 n 台计算机组成虚拟机的服务率为 $1/n\mu$。

同时假设服务实例运行环境的切换代价较大。例如，需要很长时间才能把用户商业数据从外存读入缓存。这样，每个服务实例的请求不能在虚拟机之间自由切换。每个由虚拟机和客户请求组成的服务实例用先入先出的 M/M/1 队列模拟。定义服务强度 ρ 为到达率与服务率的比例：

$$\rho=\lambda/\mu \tag{1}$$

3.2 基于平均响应时间的价格机制 MRT

价格机制详细描述计费策略，通常会在 SLA 中明确定义。定价可考虑提交时段(是否高峰)、可用资源量等因素[2]。但这些机制均面向服务提供者。本文提出面向客户的价格机制 MRT，其中实际价格基于平均响应时间。

平均响应时间是评估服务性能的常用指标。本文定义响应时间为从请求到达系统到服务完成的时间段(忽略链路延迟)。在 M/M/1 队列中响应时间也称为逗留时间(sojourn time)。由于服务实例请求到达率会随时间变化，因此有必要把时间分成片，单独计算每个时间片的平均响应时间。

首先定义一种基于平均响应时间的价格模型，称为 MRT (Mean Response Time)。用 F 代表实际响应时间 r 对基准时间 R 的偏离程度：

$$F=r/R \tag{2}$$

式中，r 是实际测量的平均响应时间，R 是 SLA 定义的性能基准。性能基准一般是指实际应用对性能的要求，服务实例之间的基准不同，如电子商务交易的基准要求一般为 2～4 秒。该模型也称为服务需求驱动模型[3]。

这样，价格机制 MRT 可以描述为：

$$B=b(1-F) \tag{3}$$

式中，B 为每次服务的价格，b 为价格常数。如图 1 所示，实际上价格 B 为平均响应时间 r 的线性函数。当平均响应时间大于阈值 R 时，服务提供商要受惩罚。图 1 同时表明，b/R 实际上就是价格函数斜率：

$$q=b/R \tag{4}$$

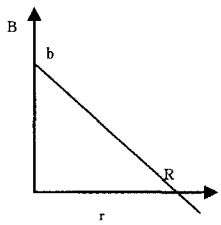

图 1 MRT 价格模型

4 收入最大化云资源分配

根据排队论 M/M/1 模型研究结论，服务实例 i 在稳定状态的平均响应时间为：

$$r_i=\frac{1}{n_i\mu_i-\lambda_i} \tag{5}$$

那么根据式(2)，服务性能等级 F_i 为：

$$F_i=\frac{1}{(n_i\mu_i-\lambda_i)R_i} \tag{6}$$

根据式(3)，一次服务的平均收入 g_i 为：

$$g_i=b_i(1-\frac{1}{(n_i\mu_i-\lambda_i)R_i}) \tag{7}$$

服务实例 i 在单位时间带来的总收入为：

$$G_i=\lambda_i g_i=\lambda_i b_i(1-\frac{1}{(n_i\mu_i-\lambda_i)R_i}) \tag{8}$$

这样，建立了单位时间内服务收入与到达率、服务率、定价机制、性能要求等参数之间的函数关系。所以，云资源分配可以形式化描述为以下最优化问题：

$$Objective: \text{Max} \sum_{i=1}^{m}\lambda_i b_i(1-\frac{1}{(n_i\mu_i-\lambda_i)R_i})$$

$$\text{s. t.} \sum_{i=1}^{m}n_i=N \tag{9}$$

下面用拉格朗日乘子法求解。构造拉格朗日复合函数：

$$L(n_i)=\sum_{i=1}^{m}\lambda_i b_i(1-\frac{1}{(n_i\mu_i-\lambda_i)R_i})+\bar{\lambda}(N-\sum_{i=1}^{m}n_i) \tag{10}$$

其中，$\bar{\lambda}$ 为常数。

令 $dL/dn_i=0, i=0,1,2,\cdots,m$：

$$\frac{\lambda_i b_i}{R_i}\frac{\mu_i}{(n_i\mu_i-\lambda_i)^2}-\bar{\lambda}=0 \tag{11}$$

$$n_i=\sqrt{\frac{1}{\bar{\lambda}}}\sqrt{q_i\rho_i}+\rho_i \tag{12}$$

将式(12)代入式(9)中的约束条件：

$$N=\sqrt{\frac{1}{\bar{\lambda}}}\sum_{j=1}^{m}\sqrt{q_j\rho_j}+\sum_{j=1}^{m}\rho_j \tag{13}$$

$$\sqrt{\frac{1}{\bar{\lambda}}}=\frac{N-\sum_{j=1}^{m}\rho_j}{\sum_{j=1}^{m}\sqrt{q_j\rho_j}} \tag{14}$$

将式(14)代入式(12)，可得到最终结论。即：

$$n_i=\frac{N-\sum_{j=1}^{m}\rho_j}{\sum_{j=1}^{m}\sqrt{q_j\rho_j}}\sqrt{q_i\rho_i}+\rho_i \tag{15}$$

然而，式(15)成立的前提条件是式(5)成立。根据排队论 M/M/1 模型研究结论，只有服务请求到达率低于服务率时式(5)才成立。否则，队列越来越长，响应时间越来越长，平均响应时间不会收敛。所以，结论式(15)成立，当且仅当到达率低于服务率：

$$\lambda_i<n_i\mu_i \tag{16}$$

$$n_i>\rho_i \tag{17}$$

此外，图 1 表明，如果平均响应时间不能满足服务需求 R_i，服务提供商会遭受惩罚性收费。所以，分配策略需保证平均响应时间不能低于 R_i：

$$r_i=\frac{1}{n_i\mu_i-\lambda_i}<R_i \tag{18}$$

$$n_i>\frac{1}{\mu_i R_i}+\rho_i \tag{19}$$

表达式(17)和式(19)给出了每个服务实例分配计算资源数量下限。同时易见，式(19)成立时式(17)肯定成立。

5 性能评估

下面以实验验证本文云资源分配理论最优算法的有效性。实验中分别用合成数据和网络日志数据模拟了请求到达率。

我们用 c 语言编程，实现了以时间驱动的模拟器。时间粒度为毫秒。每过 1 毫秒，检查处理在当前时间槽内发生的事件。事件类型共包括 4 类：到达、离开、资源再分配、输出实验结果。

本文工作的目标是实现云服务提供商服务收入最大化，所以服务收入作为实验评价的核心指标。截至目前，Michele 学位论文中提出的云资源最优分配算法与本文工作最为接近[4]，文中提出了基于服务强度的启发式算法，以此作为比较对象。

接下来分别用 MRT 和 Heuristic 代表 MRT 价格模型下的最优化算法和文献[4]中给出的最优化算法。相关实验参数和缺省值如表 1 所列。

表 1 实验参数及其缺省值

Parameter	Default Value
Arrival Rate λ	Random (20…30)
Intercept b	20/60
Number of customers m	20
Service rate μ	10
Intercept R	Random (2…32)/60
Revenue unit	$

5.1 基于合成数据的实验

在本实验中，把时间分割为片，每过一个时间片计算输出该时间片内的收入。总运行时间为 1 小时。实验结果取自最后一个时间片。

图 2 是 MRT 分配算法和 Heuristic 分配算法在不同计算能力下的收入比较。时间片设定为 30 分钟。图 2 表明，当云计算中心的计算资源增多时，获得的收入也增多，MRT 分配算法总是优于 Heuristic。这是因为 MRT 是理论上的最优算法。此外，当资源总量相对较少时，MRT 的优势尤为明显。因此，MRT 在资源数量相对有限或服务请求相对较多时更能体现其价值。

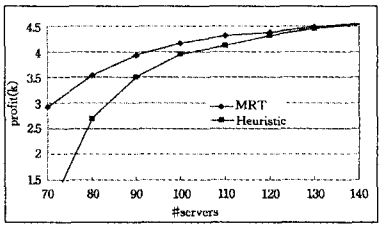

图 2 不同计算资源下 MRT 的收入比较

图 2 同时表明，Heuristic 性能同样处于较高水平，尤其当计算资源较多时性能更接近 MRT。表达式(15)可以解释该现象。该式表明，最优分配实际上分为两个步骤。首先，每个服务实例分配 ρ 的计算资源。然后，根据 ρ 和 q 分配剩余资源。这样，服务强度 ρ 在最优分配过程中占有主导地位。这便是 Heuristic 和 MRT 性能接近的原因。实际上，当价格函数斜率 q 与服务强度 ρ 具有相同分布时，Heuristic 符合最优分布。图 3 同样可验证这一观点。

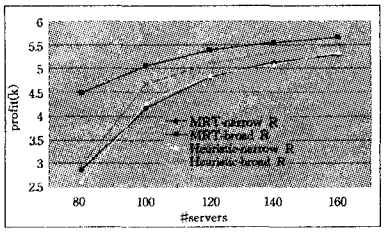

图 3　不同价格参数的收入比较

图 3 是不同价格函数斜率分布下的服务租赁收入比较。每个服务实例具有相同的 ρ,b,但具有不同的响应时间需求 R。图 3 显示,当计算资源较少时,MRT 收入上升迅速。当截距 b 固定、R 波动幅度较大时,q 的变化范围也大。效率 q 在本文最优化算法 MRT 中对最终分配具有影响,但在 Heuristic 中对分配没有影响。所以,当斜率 q 由较小波动幅度到较大波动幅度时,MRT 分配收入会比 Heuristic 分配收入增长迅速。

5.2　基于日志数据的实验

本实验采用网络抓取的数据模拟访问请求到达。使用网络公开的 Web 应用访问数据代表公共云访问请求到达率[5]。所有访问都是对 www 服务器的 HTTP 请求。我们截取连续 8 个小时的访问记录模拟云服务请求到达。用 Web 访问数据模拟云服务请求具有合理性,因为很多云服务实际上便是以 Web 服务器方式提供的。详细的实验数据集合信息如表 2 所列。

表 2　实验数据集的详细信息

#	source	Date	time	# records
1	EPA-HTTP	30 Aug. 1995	09:00-17:00	31385
2	EPA-HTTP	30 Aug. 1995	16:00-24:00	14714
3	SDSC-HTTP	22 Aug. 1995	09:00-17:00	15479
4	SDSC-HTTP	22 Aug. 1995	16:00-24:00	7178
5	NASA-HTTP	01 Jul. 1995	00:00-08:00	16481
6	NASA-HTTP	01 Jul. 1995	09:00-17:00	24021
7	NASA-HTTP	01 Jul. 1995	16:00-24:00	25476
8	NASA-HTTP	25 Jul. 1995	00:00-08:00	9360
9	NASA-HTTP	25 Jul. 1995	09:00-17:00	34965
10	NASA-HTTP	25 Jul. 1995	16:00-24:00	20652

把 8 个小时运行时间分为 5 分钟长度的时间片。在运行期间,分别计算每个服务实例每个时间片服务请求的到达数量。然后根据前一时间片和当前时间片的服务请求实际到达数量,预测下一时间片服务请求的平均到达数量。计算算法公式如下:

$$\lambda_{next}=\lambda_{current}+0.5(\lambda_{current}-\lambda_{previous}) \tag{20}$$

式中,$\lambda_{previous}$ 和 $\lambda_{current}$ 代表过去两个时间片服务请求的实际数量,通过计数易得。

云计算中心计算资源分为 10 组,每组有一个服务请求 FIFO 队列。每个时间片结束,系统会在服务实例之间重新分配计算资源,但不会立刻调整资源数量,而是经过一个时间片之后再执行。其原因在于,当前的队列长度是前一个时间片的后果。具体参数如表 3 所列。

表 3　基于日志数据实验的参数和缺省值

Parameter	Default Value
Service Time Distribution	Negative Exponential
Service Rate μ	Random (10…15)
Number of Customers m	10
Intercept b	Random (10…20)/60
Intercept R	Random (10…30)/60 mins

图 4 是服务收入随时间的演化曲线,时间单位为 5 分钟。实验中共部署 80 台计算机。MRT 和 Heuristic 的收入变化趋势与到达率有密切关系。图 4 表明,MRT 性能要优于 Heuristic。8 小时运行中,前者每 5 分钟的平均收入为392.76,后者平均收入为 334.07。前者比后者收入高17.57%。

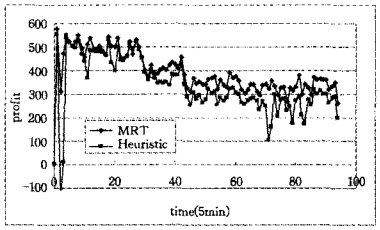

图 4　MRT 收入随时间的演化曲线

同时,图 4 表明,MRT 收入在第 47 个时间片迅速下降。我们认为,这是由到达率的剧烈波动造成的。如图 5 所示,到达率从第 42 到第 45 时间片下降明显,但从第 46 时间片迅速上升。因此,根据式(20)预测的到达率与实际到达率会有较大偏差,从而误导资源分配。到达率从第 21 时间片到第 25 时间片上升明显,而在 25 至 31 时间片急转直下。相同原因,MRT 收入在第 26 时间片均下降明显。所以,高质量的服务请求到达预测算法对资源分配效果具有积极影响。

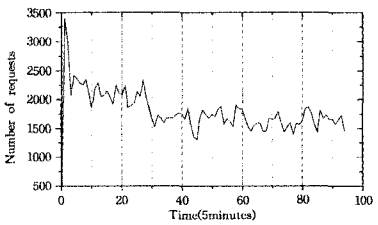

图 5　服务请求到达随时间的演化曲线

结束语　具有法律效力的服务级别协议是促成云模型中用户和服务提供商之间合作的重要桥梁。本文着重阐述服务提供商如何根据 SLA 中基于性能的价格模型,动态调度云中心资源池,使得云服务收入最大。同时需要指出,通过调整价格函数,本文工作能够以非常灵活方便的形式实现资源的优化管理。

本文对资源分配优化问题进行形式化描述,然后借助拉格朗日乘子法给出了最优解。实验结果表明,该算法优于相关工作;在云资源相对短缺或接入服务实例较多时,本文算法的优势更加明显。

参 考 文 献

[1] Gong Chun-ye, Liu Jie, Zhang Qiang, et al. The Characteristics of cloud Computing[C]// Proc. of 39th International Conference on Parallel Processing Workshops. 2010:275-279

[2] Buyya R, Yeo C S, Venugopal S, et al. Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility[J]. Future Generation Computer Systems, 2009, 25:599-616

[3] Zhu Hui-can, Tang Hong, Yang Tao. Demand-driven Service Differentiation in Cluster-based Network Servers[C]// Proc. IEEE INFOCOM 2001. 2001:679-688

[4] Mazzucco M. Revenue Maximization Problems in Commercial Data Centers[D]. University of Newcastle, 2009

(下转第 61 页)

参 考 文 献

[1] Akyildiz I F, Melodia T, et al. A survey on wireless multimedia sensor networks[J]. Computer Networks, 2007, 51: 921-960

[2] Bhanu B, Ravishankar C V. Distributed Video Sensor Networks [M]. Springer-Verlag, 2011

[3] Murat U, Sclarooff S. Event prediction in a hybrid camera network[J]. ACM Transactions on Sensor Networks, 2012, 3(8): 1-16

[4] Song Bi, Morye A. Collaborative Sensing in a Distributed PTZ Camera Network[J]. IEEE Transactions on Image Processing, 2012, 21(7): 3282-3295

[5] Lobaton E. A Distributed Topological Camera Network Representation for Tracking Applications[J]. IEEE Transactions on Image Processing, 2010, 19(10): 2516-2529

[6] 南国芳,陈忠楠. 基于进化优化的移动感知节点部署算法[J]电子学报,2012,40(5):1017-1022

[7] 陶丹,马华东. 有向传感器网络覆盖控制算法[J]. 软件学报,2011,22(10):2317-2334

[8] 陶丹,孙岩,陈后金,视频传感器网络中最坏情况覆盖检测与修补算法[J]. 电子学报,2009,37(10):2284-2290

[9] 肖甫,王汝传,叶晓国,等. 基于改进势场的有向传感器网络路径覆盖增强算法[J]. 计算机研究与发展,2009,46(12):2126-2133

[10] 蒋一波,王万良,陈伟杰. 视频传感器网络中无盲区监视优化[J]. 软件学报,2012,23(2):310-322

[11] Xu Yi-chun, Lei Bang-jun, Hendriks E A. Camera Network Coverage Improving by Particle Swarm Optimization[J]. EURASIP Journal on Image and Video Processing, 2011, 1(1): 3

[12] Conci N, Lizzi L. Camera Placement Using Particle Swarm Optimization in Visual Surveillance Applications[C]// 2009 16th IEEE International Conference on Image Processing (ICIP). Cairo, 2009: 3485-3488

[13] Zhou Pu, Long Cheng-nian. Optimal Coverage of Camera Networks Using PSO Algorithm[C]// 4th International Congress on Image and Signal Processing. Shanghai, 2011: 2084-2088

[14] Aouf N, Djouadi M S. Particle Swarm Optimization Inspired Probability Algorithm for Optimal Camera Network Placement [J]. IEEE Sensors Journal, 2012, 12(5): 1402-1412

[15] Lee D, Lin A. Computational complexity of art gallery problems [J]. IEEE Transactions on Information Theory, 1986, 32(2): 276-282

[16] 刘志刚,曾嘉俊,韩志伟. 基于个体最优位置的自适应变异扰动粒子群算法[J]. 西南交通大学学报,2012,47(5):761-768

[17] 林川,冯全源. 粒子群优化算法的信息共享策略[J]. 西南交通大学学报,2009,44(3):437-441

[18] Mouzon G, Yildirim M B. Genetic algorithm to solve a multi-objective scheduling problem[C]// Eichhorn DM, ed. Proc. of the 3rd Annual GRASP Symp. Wichita: Wichita State University, 2007: 45-46

[19] van den Bergh F. A Cooperative approach to particle swarm optimization [J]. IEEE Transactions on Evolutionary Computation, 2004, 8(3): 225-239

[20] Chen Chao-hong, Chen Ying-ping. Convergence time analysis of particle swarm optimization based on particle interaction[J]. Advances in Artificial Intelligence, 2011(1): 7

(上接第 39 页)

[5] The Internet Traffic Archive[OL]. http://ita.ee.lbl.gov/html/traces.html, 2011-08-23

[6] Walsh W E, Tesauro G, Kephart J O, et al. Utility Functions in Autonomic Systems[C]// Proc. of the International Conference on Autonomic Computing. 2004: 70-77

[7] Rajkumar R, Lee C, Lehoczky J, et al. A Resource Allocation Model for QoS Management[C]// Proc. of the 18th IEEE Real-Time Systems Symposium. 1997: 298-307

[8] Ghosh S, Rajkumar R, Hansen J, et al. Scalable Resource Allocation for Multi-processor QoS Optimization[C]// Proc. of 23rd International Conference on Distributed Computing Systems. 2003: 174-183

[9] Hansen J P, Ghosh S, Rajkumar R, et al. Resource Management of Highly Configurable Tasks[C]// Proc. 18th International Parallel and Distributed Processing Symposium. 2004: 116

[10] Menascé D A, Barbará D, Dodge R. Preserving QoS of E-Commerce Sites Through Self-Tuning: A Performance Model Approach[C]// Proc. of 3rd ACM conference on Electronic Commerce. 2001: 224-234

[11] Bennani M N, Menascé D. Resource Allocation for Autonomic Data Centers Using Analytic Performance Models[C]// Proc. of the Second International Conference on Autonomic Computing. 2005: 229-240

[12] Chandra A, Gong Wei-bo, Shenoy P. Dynamic Resource Allocation for Shared Data Centers Using Online Measurements[C]// Proc. of the 11th IEEE/ACM International Workshop on Quality of Service. 2003: 381-400

[13] Levy R, Nagarajarao J, Pacifici G, et al. Performance Management for Cluster Based Web Services[C]// Proc. of IEEE 8th International Symposium on Integrated Network Management. 2003: 247-261

[14] Li Ying, Sun Ke-wei, Jie Qiu, et al. Self-reconfiguration of Service-based Systems: A Case Study for Service Level Agreements and Resource Optimization[C]// Proc. of IEEE International Conference on Web Services. 2005: 266-273

[15] Zhang Li, Ardagna D. SLA Based Profit Optimization in Autonomic Computing Systems[C]// Proc. of the 2nd International Conference on Service Oriented Computing. 2004: 173-182

[16] Liu Zhen, Squillante M S, Wolf J L. On Maximizing Service-Level-Agreement Profits[C]// Proc. of the 3rd ACM Conference on Electronic Commerce. 2001: 213-223

[17] Tim Püschel, Nikolay Borissov, Dirk Neumann, et al. Extended Resource Management Using Client Classification and Economic Enhancements[C]// Proc. of eChallenges Conference. 2007: 65-72

[18] Villela D, Pradhan P, Rubenstein D. Provisioning Servers in the Application Tier for E-Commerce Systems[J]. ACM Transactions on Internet Technology, 2007, 7(1): 57-66