

一种基于性能评估的元任务调度算法

胡周君, 胡志刚, 李林

(中南大学信息科学与工程学院, 410083, 长沙)

摘要: 在保障本地任务 QoS 需求及尊重网格节点的本地调度策略的条件下, 提出了一种资源动态性能评估方法, 并得出了 2 个资源动态性能的评估尺度. 基于网格资源可用性的相关结果, 利用概率法得出网格节点服务于网格任务的平均能力. 利用热力学中熵的概念对网格节点状态的动态变化不确定性进行了评估, 并得到刻画网格节点状态不确定性的定量值. 在资源动态性能评估的基础上, 提出了一种基于性能评估的元任务调度算法. 仿真与评估结果表明, 所提算法在服务拒绝率、网格任务执行跨度方面优于以往算法, 且在网格用户的 QoS 保障程度上有所增强.

关键词: 服务网格; 性能评估; QoS 保障; 调度算法

中图分类号: TP393 **文献标志码:** A **文章编号:** 0253-987X(2008)08-0972-05

Performance-Evaluation-Based Meta-Task Scheduling Algorithm

HU Zhoujun, HU Zhigang, LI Lin

(School of Information Science and Engineering, Central South University, Changsha 410083, China)

Abstract: Providing non-trivial QoS for grid application under meeting local job's QoS requirements and local scheduling strategy of grid site, is an important objective and a new issue of task scheduling in service grid environment. A resource dynamic performance evaluation method is presented and two performance metrics are obtained. Based on resource availability characteristic of a grid site, probability theory is employed to evaluate the average capacity of a resource. Several states for grid sites are labeled, and the conception of entropy in thermodynamics is used to describe the uncertainty of the resource. Then, a performance-evaluation-based meta-task scheduling algorithm is proposed. On the basis of extended and revised gridSim, the algorithm is implemented and compared with two related existing algorithms. The experiment results show that the proposed algorithm outperforms the existing ones on both aspects of the service deny rate and the grid task makespan.

Keywords: service grid; performance evaluation; QoS guarantee; scheduling algorithm

计算服务网格^[1]是一种大规模的网络计算系统, 它将开放式网格服务体系结构(OGSA)与 Web 服务资源框架(WSRF)相结合, 以“服务”的方式来封装多机构虚拟组织中各种高性能计算资源, 并统一以网格服务的形式提供给用户使用. 在服务网格中, 面向 QoS 的“区分确保”型^[2]任务调度法是以保障用户 QoS 需求为目标的, 它取代了传统的以最小化任务完成时间为目标的“尽力而为”型网格任务调

度法^[3]. 目前, 大多数任务调度算法^[4-6]是基于资源的静态性能信息, 根据用户 QoS 需求制定调度决策的. 静态任务调度算法难以适应动态服务网格的环境要求, 而在设计动态的任务调度算法时又存在 2 个重要问题: 如何同时保障本地任务与网格任务的 QoS 需求; 如何对本地负载动态变化的网格节点的服务能力进行评估.

针对上述 2 个问题, 本文基于对网格节点相关

资源可用性的历史数据进行统计、分析得出的结果,在保障本地任务 QoS 需求、尊重自治网格节点本地调度策略的前提下对资源动态性能进行了评估,据此提出了一种基于性能评估的元任务调度算法,以保障服务网格中的用户 QoS 需求。

1 相关工作

目前,基于资源性能评估、预测的调度系统主要有 2 种,即 GHS^[7]和 EMPEROR^[8]. GHS 将网格节点处理提交任务视为一个排队系统,这样通过对相关历史数据的统计分析,依据排队理论便可得到任务完成时间的概率分布函数.可以看出,这种结果只是预测任务的完成时间不超过某一值的概率,并不是具体的预测值,因此存在不确定性. EMPEROR 是基于时间序列分析模型对资源的 CPU、内存负载进行预测,从而获得任务的完成时间,该方法计算量大且其远期预测效果较差.文献[9]是通过多个历史网格任务在一个节点上的实际执行时间与无负载执行时间的比值,来估计即将调度的任务在该节点上的实际执行时间,并基于网格任务在资源能力动态变化的网格节点上的实际执行时间,对 MCT 和 Min-min 算法进行了改进,但该预测方法缺乏合理依据.文献[10]对动态关键路径算法进行了改进,即根据网格资源空闲时间槽(time slot)及任务等级提出了一种资源性能波动感知的工作流调度算法.该算法仅利用了资源空闲时间段,并且在每次调度完成一个工作流任务之后,都需要对其他未调度的工作流任务等级进行动态更新,就存在大量任务的网格环境来说,它的计算复杂度比较大。

2 问题描述

在某一时间 T 内,网格节点执行若干个到达本地的任务需花费的总时间为

$$U = T_l = X_1 + X_2 + \cdots + X_n \quad (1)$$

式中: $X_i(i=1, \dots, n)$ 表示第 i 个本地任务的执行时间; n 表示时间 T 内到达的本地任务数。

假设本地任务的执行时间 $X_i(1 \leq i \leq n)$ 服从 k 阶超指数分布^[11],其概率分布函数为

$$F(x) = \sum_{i=1}^k \alpha_i F_i(x), \quad F_i(x) = 1 - e^{-\lambda_i x}, \quad \sum_{i=1}^k \alpha_i = 1 \quad (2)$$

其中参数 λ_i, α_i 是由各网格节点根据本地任务执行时间的历史数据统计获得的,并且可以动态更新。

将网格节点处理本地任务视为一个排队系

统^[7],根据排队理论可知,在时间段 Δt 内,本地任务到达数 N 为一随机变量且服从参数为 γ 的泊松分布,则在时间 T 内的本地任务到达数为 n 的概率为

$$P\{N = n\} = \gamma^n e^{-\gamma} / ((\frac{\Delta t}{T} n)!) \quad (3)$$

其中参数 $\gamma, \Delta t$ 是由各网格节点根据本地任务到达数的历史数据统计获得的,并且可以动态更新。

3 资源动态性能评估

定理 1 设随机变量 $X_1, X_2, \dots, X_N$ 相互独立且均服从 k 阶超指数分布(见式(2)), N 为一随机变量,服从参数为 γ 的泊松分布, X 与 N 相互独立,则随机变量 $U = \sum_{j=1}^N X_j$ 的分布函数为

$$F(u, n) = \begin{cases} \gamma^n e^{-\gamma} / ((\frac{\Delta t}{T} n)!) \sum_{i=1}^k \alpha_i \left\{ -\frac{1}{\lambda_i} \left\{ e^{-\lambda_i u} u^{n-1} + \frac{n-1}{\lambda_i} \left[e^{-\lambda_i u} u^{n-2} + \cdots + \left(\frac{3}{\lambda_i} (e^{-\lambda_i u} u^2 + \frac{2}{\lambda_i} (e^{-\lambda_i u} u + \frac{1}{\lambda_i} (e^{-\lambda_i u} - 1))) \right) \right] \right\} \frac{\lambda_i^n}{(n-1)!} \right\} & u \geq 0 \\ 0 & u < 0 \end{cases} \quad (4)$$

由于篇幅有限,分布函数证明过程略去。

在某一时间 T 内,根据节点资源的可用能力可将一个网格节点分为若干种定性描述的状态 $S_1, S_2, \dots, S_m$,如果网格节点在该时间内处理本地任务的总时间 $t_l \in [0, T]$ 服从一定的概率分布,那么该网格节点可以隶属度 p_i 隶属于状态 $S_i(1 \leq i \leq m)$. 为了便于理解,本文将形式化描述实例化,例如:根据网格节点执行本地任务的时间情况,可在 10 h 内将网格节点状态分为空闲、较空闲、一般、繁忙、很繁忙等 5 种情况,各状态对应本地任务的执行时间范围分别为 $S_1: [0, 2], S_2: [2, 4], S_3: [4, 6], S_4: [6, 8], S_5: [8, 10]$,各状态相应的取值依据 $S_i = i$ 。

根据定理 1,网格节点隶属于一种状态的概率

$$p_i = F(b, \bar{n}) - F(a, \bar{n}) \quad (5)$$

式中: a, b 分别为状态 S_i 对应时间范围的左、右界; $\bar{n}$ 为本地任务到达数的均值,即 $\bar{n} = \gamma T / \Delta t$ 。

本文将用以下 2 个评估尺度对资源动态性能进行评估,即

$$E = \sum_{i=1}^m S_i p_i \quad (6)$$

$$S_E = - \sum_{i=1}^m p_i \ln p_i \quad (7)$$

E 反映了网格节点处理本地任务的平均繁忙程度,为网格节点所处状态的偏好水平,该值越大,网格节点越繁忙,网络任务的执行时间就越少;熵 S_E 反映了网格节点状态的集中程度,为网格节点状态动态变化的离散程度,该值越大,网格节点状态就越不稳定。

资源动态性能评估是建立在各网格节点对本地任务的达数、执行时间进行周期性监控基础上的,首先对监控数据进行统计、分析,然后利用分布模型对资源可用性特性进行刻画,再依据资源可用性的分布特性,通过资源性能评估便可得出节点的平均状态以及节点状态不确定性的度量值,最后对度量值进行服务封装。在服务注册时,可在统一的服务描述语法中增加服务性能评估结果的描述信息。监控与评估应在提供服务的资源端进行,这样就无需对资源进行集中管理,从而大大缓解了中间件系统的压力。

4 基于性能评估的网格任务调度算法

截止时间和价格是用户通常考虑的 2 个 QoS 需求,对于大规模的科学应用,用户往往更关心的是任务的完成时间,所以,本文基于资源动态性能评估结果,将网格任务调度到真正能够满足用户的截止时间要求的资源上,以降低服务拒绝率,保障用户 QoS。本文的调度算法思想如下。

首先,通过资源发现得到一个候选网格节点集,对于该集合中的每一个网格节点,检查其价格是否满足用户要求,再根据节点所处的平均状态情况及资源能力判断是否满足用户的截止时间要求,该判断式为

$$\sum_{i=1}^m T_{Mi} p_i + l(j_i)/c \leq T_D \quad (8)$$

式中: T_{Mi} 为网格节点各状态 S_i 对应本地任务执行时间的中值; $l(j_i)$ 为任务 j_i 的长度; c 为网格节点静态计算能力; T_D 为任务 j_i 的截止时间。

其次,在同时满足以上 2 个需求的所有网格节点中选取熵值(即状态不确定性)最小的节点,对于熵值相同的节点,选择任务完成时间最少的节点,作为调度的目标节点。

基于性能评估的网格任务调度算法所示的伪码如图 1 所示。该算法通过对网格环境下的资源动态能力评估来确定调度目标资源,其复杂度为 $O(n)$,又因该算法主要是从资源的平均服务能力和稳定程

```

输入: 任务集  $S$ , 截止时间  $D$ , 价格  $C$ , 候选网格节点集  $R$ 
输出: 任务映射方案
1. 初始化: 集合  $map$  保存任务-节点映射对, 变量  $i, k$ 
   用于计数;
2. for task  $t_i$  in  $S$ , do
3. Repeat
4.   for resource  $R_i$  in  $R$ 
5.     if 用户出价  $C_i$  小于等于节点价格  $C_{R_i}$ 
       and  $\sum_{i=1}^m T_{Mi} p_i + l(j_i)/c \leq T_D$ 
6.       将节点  $R_i$  加入到调度节点集合  $R_S$  中;
7.     Endif
8.   Endfor
9. 找出集合  $R_S$  中熵值最小的节点  $\min(S_{ER_k})$ , 返回  $R_k$ ;
10. if 返回多个  $R_k$ , 选择  $j_i$  完成时间最小的  $R_k$ 
11.   将任务  $j_i$  映射到节点  $R_k$  上, 即  $map(j_i, R_k)$ ;
12.   remove task  $j_i$  from  $S$ 
13. Endfor
14. return  $map(t_i, R_k)$ 

```

图 1 基于性能评估的元任务调度算法

度上进行资源评估的,所以在任务执行时间上的优势比较明显。调度时,算法只是将价格作为选择资源的一个条件,并没有对价格 QoS 进行优化。

5 实验结果与性能评估

5.1 仿真实验设计

本文对 GridSim^[12] 进行了改进与扩展,主要包括网格节点的本地任务提交过程以及本地调度策略,同时将本文的资源性能评估机制融入 GridSim。改进与扩展的内容如下:①设计了一个类模拟网格节点的本地任务提交过程,参数 $\gamma, \lambda_i (i=2), \alpha_1 (\alpha_2 = 1 - \alpha_1)$ 符合现实统计得出的结果;②本地任务调度包括抢占式和时间片轮转式 2 种策略;③网格节点对象扩展的接口包括设定或通过动态性能评估获取的资源平均状态与熵值、资源就绪时间、本地调度策略等。

下面,用改进与扩展后的 GridSim 来验证本文算法,并与调度器 EMPEROR^[8] 中基于预测的调度算法(算法 1)及 Sun 等人提出的 Min-min 启发式算法(算法 2)^[13],就网格任务执行跨度和服务拒绝率进行比较。

实验场景:将本文算法与不同预测精度的算法 1 进行比较,将本文算法与不同本地负载的算法 2 进行比较,同时检验不确定性状态条件对本文算法的影响。对于每一个实验场景重复 50 次,取其平均结果作为实验结果。

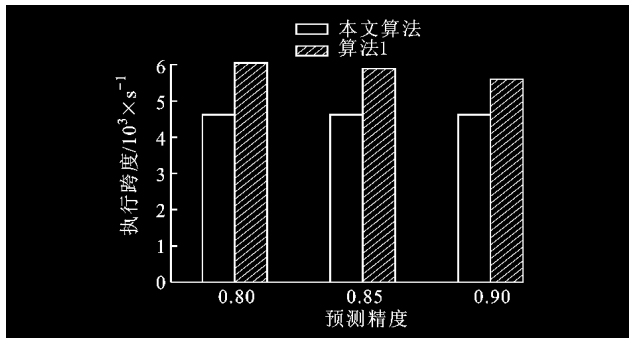
模拟的网络环境包括:3 个网格用户,每个用户提交 30 个网格任务,任务长度在 $[2, 40] \times 10^{11}$ 指令

数范围内随机产生;100个网格资源,资源静态能力在 $[2,3] \times 10^9$ 指令数/s范围内随机产生;参数 γ, λ_i ($i=2$)、 α_1 ($\alpha_2=1-\alpha_1$)分别在指定范围 $[0.5, 5]$ 、 $[0.1, 0.6]$ 、 $[0, 1]$ 之内随机产生。

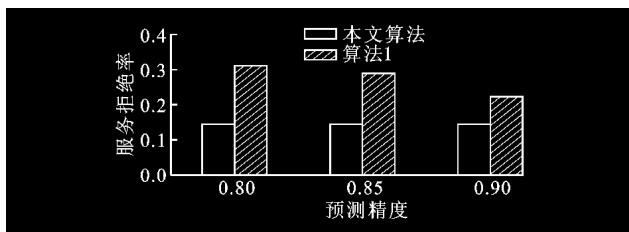
5.2 算法性能评估

(1)与不同预测精度的算法1比较. 预测精度分别为0.8、0.85、0.9时,本文算法与算法1的执行跨度和服务拒绝率比较如图2所示. 从中看出,当预测精度为0.8时,本文算法优势明显,因为在满足用户QoS条件下,本文算法将任务调度到了状态最稳定的节点上,而算法1却未考虑网格节点状态的波动性. 随着预测精度的提高,2种算法的执行跨度逐渐接近. 若假设预测精度接近1,算法1的性能优势可能胜出,但其平均预测精度仍不足0.9,可见这种假设是不成立的。

在服务拒绝率方面,本文算法同样优势明显,但部分服务被拒绝的主要原因有二,其一所提性能评估方法是一种基于历史数据的统计方法,其二用户提出的截止时间的QoS要求过于严格. 因此,本文算法在执行跨度与服务拒绝率这2个方面均优于以往算法。



(a) 执行跨度

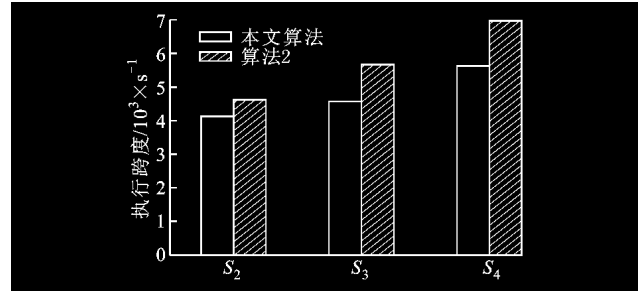


(b) 服务拒绝率

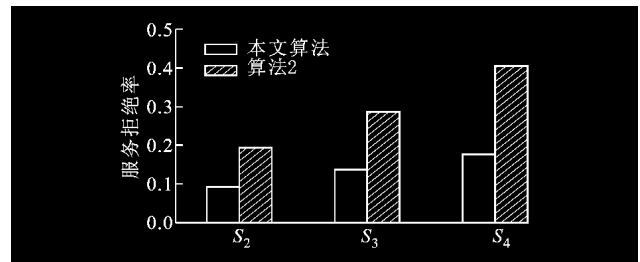
图2 不同预测精度下本文算法、算法1的性能比较

(2)与不同本地负载的算法2比较. 网格节点在 S_2 (较空闲)、 S_3 (一般)、 S_4 (繁忙)3种状态(对应不同的本地负载)下,本文算法与算法2的执行跨度和服务拒绝率比较如图3所示. 从中看出,当本地负载

较重时(S_4),本文算法优势比较明显,因为算法2未考虑网格节点的本地负载情况. 随着本地负载逐渐减轻,2种算法的性能差距逐渐减小. 若假设网格节点无本地负载,则2种算法的性能相同. 实际上,网格节点存在着本地负载,且是动态变化的,因此本文算法更符合实际情况。



(a) 执行跨度



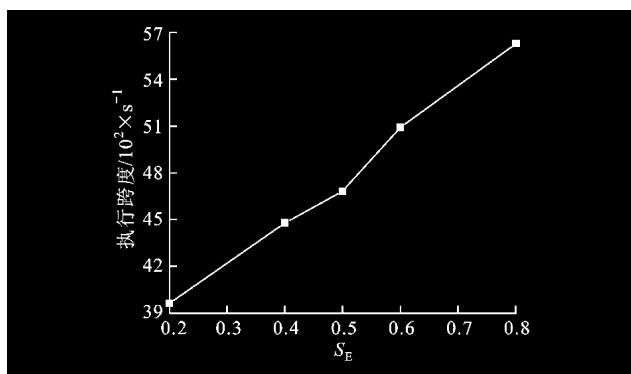
(b) 服务拒绝率

图3 不同本地负载下本文算法、算法2的性能比较

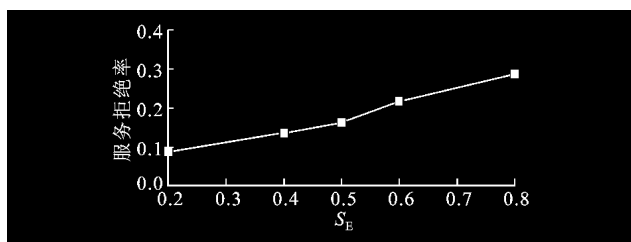
(3)熵值对算法性能的影响. 本文算法在网格节点状态固定为 S_3 (一般)的情况下,熵值取0.2、0.4、0.5、0.6、0.8时的执行跨度和服务拒绝率如图4所示. 从中看出,网格节点状态的不确定性影响着执行跨度和服务拒绝率. 随着熵值的逐渐增大,执行跨度和服务拒绝率越来越大. 因为熵值可反映系统的不确定性,状态越不确定的网格节点会导致执行跨度和服务拒绝率值越大,甚至平均状态繁忙但稳定的节点可能比平均状态较空闲但不稳定的节点所产生的执行跨度和服务拒绝率值要小. 因此,在满足用户QoS要求的网格节点中,通过选择熵值最小的网格节点,可大大增强网格用户QoS的保障程度。

6 结束语

本文在保障本地任务QoS需求并尊重自治网格节点本地调度策略的前提下,基于网格资源可用性的相关结果,利用概率论方法对网格节点的资源能力进行了评估. 通过对网格节点状态的划分,给出了资源的平均状态,并利用热力学中熵的概念对资



(a) 执行跨度



(b) 服务拒绝率

图4 不同熵值对本文算法的影响

源动态变化的不确定性进行了刻画。在此基础上,提出了一种基于性能评估的元任务调度算法,同时基于改进与扩展后的 GridSim 对所提算法进行了仿真。结果表明,本文算法不但增强了网格用户 QoS 的保障程度,而且降低了网格任务的执行跨度。今后的工作主要是进一步优化本地任务 QoS 和网格任务 QoS 的保障力度,并在设计任务调度算法时充分考虑任务间的通信因素等。

参考文献:

- [1] FOSTER I, KESSELMAN C, NICK M, et al. The physiology of the grid: an open grid services architecture for distributed systems integration [EB/OL]. [2006-12-10]. <http://www.globus.org/research/papers/ogsa.pdf>.
- [2] 胡春明, 怀进鹏. 一种支持端到端 QoS 的服务网格体系结构 [J]. 软件学报, 2006, 17(6): 1448-1458.
HU Chunming, HUAI Jinpeng. A service oriented grid architecture with end to end quality of service [J]. Journal of Software, 2006, 17(6): 1448-1458.
- [3] SUN Xianhe, WU Ming. Quality of service of grid computing: resource sharing [C] // 6th International Conference on Grid and Cooperative Computing. Los Alamitos, CA, USA: IEEE Computer Society, 2007: 395-402.
- [4] FUJIMOTO N, HAGIHARA K. A comparison among grid scheduling algorithms for independent

coarse-grained tasks [C] // Proceedings of the 2004 International Symposium on Applications and the Internet Workshops. Los Alamitos, CA, USA: IEEE Computer Society, 2004: 674-680.

- [5] TOPCUOGLU H, HARIRI S, WU M Y. Performance-effective and low-complexity task scheduling for heterogeneous computing [J]. IEEE Transactions on Parallel and Distributed Systems, 2002, 13(3): 260-274.
- [6] DONG Fangpeng, AKL S G. An adaptive double-layer workflow scheduling approach for grid computing [C] // 21st International Symposium on High Performance Computing Systems and Applications. Los Alamitos, CA, USA: IEEE Computer Society, 2007: 1-7.
- [7] WU Ming, SUN Xianhe. Grid harvest service: a performance system of grid computing [J]. Journal of Parallel and Distributed Computing, 2006, 66(10): 1322-1337.
- [8] ADZIGOGOV L, SOLDATOS J, POLYMENAKOS L. EMPEROR: an OGSA grid meta-scheduler based on dynamic resource predictions [J]. Journal of Grid Computing, 2005, 3(1): 19-37.
- [9] GAO Zhan, LUO Siwei, DING Ding. A scheduling mechanism considering simultaneous running of grid tasks and local tasks in the computational grid [C] // 2007 International Conference on Multimedia and Ubiquitous Engineering. Los Alamitos, CA, USA: IEEE Computer Society, 2007: 1100-1105.
- [10] DONG Fangpeng, AKL S G. PFAS: a resource-performance-fluctuation-aware workflow scheduling algorithm for grid computing [C] // 21st Parallel and Distributed Processing Symposium. Los Alamitos, CA, USA: IEEE Computer Society, 2007: 1-9.
- [11] MUTKA M W, LIVNY M. The available capacity of a privately owned workstation environment [J]. Performance Evaluation, 1991, 12(4): 269-284.
- [12] SULISTIO A, PODUVALY G, BUYYA R, et al. Constructing a grid simulation with differentiated network service using gridsim [C] // Proceeding of 6th International Conference on Internet Computing. Las Vegas, Nevada, USA: CSREA Press, 2005: 437-444.
- [13] HE Xiaoshan, SUN Xianhe. QoS guided min-min heuristic for grid task scheduling [J]. Journal of Computer Science and Technology, 2003, 18(4): 442-445.

(编辑 苗凌)