基于传感器的网格服务匹配问题的优化解*)

郝卫东[1] 杨 扬[1] 林 闯[2]

(北京科技大学信息工程学院 北京 100083)[1] (清华大学计算机科学与技术系 北京 100084)[2]

摘 要 在开放网格服务架构(OGSA)下网格服务匹配是当前研究的热点。通过在服务网格模型中引入监控作业运行时间的传感器,给出了基于运行时间权矩阵的网格作业-网格服务匹配的优化解,经过仿真研究其算法与其它同类算法相比可使服务匹配系统性能明显改善。本算法还给出了用户作业和服务资源之间能实现完备匹配的充分必要条件。

关键词 服务网格,传感器,匹配器,完备匹配,权矩阵

1 引言

随着计算技术和互联网络技术的发展,以网格技术为核心的新一代网络计算环境已经成为当前国际研究的一个热点和前沿领域[1]。比如,文[2]提出了一种标准的资源管理协议,文[3]较早的提出了协同服务的概念,并支持在多个管理域之间对资源使用情况进行协调,文[4]给出了一种标准的资源和任务需求描述机制并在此基础上提出了资源匹配的框架。

以开放网格服务架构 OGSA[5]为标志,网格资源管理技术变得成熟和标准化了[6,7]。OGSA 把一切(包括资源和任务)都抽象为开放的网格服务,从而为资源和任务的需求描述和发现、获取奠定了统一的机制和方法。服务网格的核心优势在于它能用统一的方式描述、发现、分配并协商使用网格可以达到的范围内的各种服务,包括计算服务、应用软件服务、网络带宽服务或存储服务等。

但是,在面向服务的体系结构下网格的资源管理(或称服务管理)仍面临着严重的挑战。首先,不同管理域组织采用不同的策略操纵它们的资源,而且资源消费者和资源提供者的目的可能不一致甚至相互抵触;其次,许多网格应用,比如电子商务或电子政务应用中常常涉及的数据传输服务,需要多个资源的并发分配,此时需要通过适当的协作调度机制,使得一组资源在同一时段内可用。

本文引入了对服务执行时间或费用的监控机制,从包含而又超越资源消费者和资源提供者的网格服务管理系统的角度提出性能指标,消除了资源消费者和资源提供者之间存在的策略和目的不一致性;本文还给出了用户任务和服务资源之间能实现完备匹配的充分必要条件;在此基础上,本文讨论了基于服务监控信息的完备匹配的最优化解。

2 基于监控的网格服务匹配模型

"资源"是一种用来满足用户作业或资源请求的可重用的实体,可以是一台机器、网络或者由机器、网络和软件联合起来而合成的某种服务。应用直接或者间接向网格提交的资源请求被称为网格的"作业"。"资源提供者"被定义为控制资源的代理。而"资源消费者"被定义为控制消费者的代理。在基于 OGSA 的服务网格系统中,资源提供者和资源消费者都是某种服务,类似地,"资源管理系统"被定义为分布式网络计算系统提供的一种服务,它负责管理一系列指定的对于网络计算可用的资源,从而实现以系统为中心的性能最优化。

当前网格资源管理中大量的行为集中在从资源提供者和消费者的角度理解并管理不同组织域内的服务策略,通常,任务处理通过作业提交来实现,而资源能力通过专门的服务质量接口来提供。比如,文[6,7]给出了基于队列优先级(head of line, HOL)和加权优先级(weighted priority, WP)等服务质量接口的方案并加以了比较。

资源发现和服务匹配是资源管理中最核心的问题。资源发现就是查询网格分布状态的过程,用于识别那些特征和状态与资源消费者的需求相符合的资源。服务匹配是从资源发现所提供的候选资源服务集合中选择和指派资源的过程,该过程主要根据一些高层应用的服务质量指标而进行,比如完成时间、可靠性或代价等。将资源发现从分配操作中分离有着重要的影响。由于发现并不意味着任何承诺,我们可以将其构建为轻量级的非权威的操作。

*)基金项目:国家自然科学基金项目(No. 90412012)。郝卫东 博士研究生,主要研究方向:计算机网络及应用,网格技术和多媒体通信等;杨 扬 教授,博士生导师,主要研究方向:表单处理和模式识别,网格技术和多媒体通信等;林 闯 教授,博士生导师,主要研究方向:计算机网络和计算机系统的性能评价网格计算等。

另一方面，当服务网格采用保证服务质量的预留策略时，服务匹配工作成了核心而基本无须资源发现操作。

考虑到资源管理不关心资源和服务的核心功能本身，而关心该功能执行的方式，比如请求的操作何时开始执行，它需要多长时间完成，它的成本或费用如何等。因此，我们建立独立于资源提供者和资源消费者的匹配服务和监控服务，形成服务网格的简化四元模型。该模型包括如下四个部分：.

Jobmanager(作业管理器)：服务网格的服务消费者代理。它作为终端用户的入口，提供作业提交、查询和删除等的处理接口。

Servicemanager(服务管理器)：服务网格的服务提供者代理。它管理资源，并负责调度合适的作业到该资源上运行。

Matchmaker(匹配器)：其功能是通过某种算法寻找消费者和提供者之间可能的匹配，并通知匹配的各方进行绑定操作。

Sensor(传感器)：其功能是实时监控资源和任务的运行状态。监控是通过被放入到资源和/或任务的传感器来记录数据的过程。

该模型的特点是增加了用于监控的 Sensor。Sensor 作为独立的网格服务，通过检测当前在资源上运行的任务的状态，比如作业启动、作业正在运行、作业挂起、作业激活、作业完成等信息，并记录相关状态事件发生的时间戳轨迹，从而刻画出系统和网络各个方面的详细性能。对于网格应用，响应时间是常常被用户关心的性能指标。本文将着重分析基于对响应时间的监控而构造的作业-服务匹配算法。

对于现在的 CPU 处理器，如 SPARC V9，Itanium 等都包含高精度的时钟，Intel Pentium 系列处理器还具有用户级指令可实现对高精度时钟的低延迟访问。考虑到网格的分布性，要检测时间还需要解决时钟同步问题。一种方法是使用外部时钟来同步，比如，使用 GPS 卫星接收器就可以获得一个高精度的时钟，在几个毫秒内完成时钟同步。另一种方法是在网格节点之间通过网络通信来同步时钟，比如采用 NTP(Network Time Protocol)协议。NTP 使用一种层次式时间服务器提供分布式时钟更新，其最高层与一些高精度的时钟源同步。每一个节点都会周期性的访问 NTP 时间服务器。通过估计时间服务器和节点之间的往返时延，NTP 算法能够返回一个带有时间戳的响应以补偿从服务器传送到客户端所消耗的时间。比如可以通过 NetLogger[8]、ARM[9]等工具实现监控。

下面从图论的角度给出网格的定义。

令无向带权偶图 $G=\langle V,E,W\rangle$，其中顶点集 $V=\{v_1,v_2,\cdots,v_n\}$，边集 $E=\{e_1,e_2,\cdots,e_m\}$，对于 G 中任意一条边 (v_i,v_j) 都存在实数 $w_{ij}\in W$ 与之对应，称 W 为权集，必可将 V 分成两个子集 V_1,V_2，并且满足 $V_1\cap V_2=\varnothing$，$V_1\cup V_2=V$，使得 G 中任意一条边 $e_{ij}(e_{ij}=(v_i,v_j)\in E)$ 的两个端点，一个属于 V_1(如 $v_i\in V_1$)，另一个属于 V_2(如 $v_j\in V_2$)。

根据上面的描述，可以给出网格的定义如下：

定义 1 定义网格 G 为无向带权偶图 $G=\langle V_1,V_2,E,W\rangle$ 的形式，其中用 V_1 表示网格作业集合，V_2 表示网格服务集合，其中，记网格作业集合的作业个数为 $p=|V_1|$，网格服务集合中的服务个数为 $q=|V_2|$，E 表示网格作业和网格服务之间可能存在的匹配，W 表示网格作业和网格服务之间某个匹配的数量指标，比如本文中赋予它作业完成时间的含义，即 $w_{ij}\in W$ 表示在服务 j 上完成作业 i 所花费的时间，其中 $i=1,2,\cdots,p$；$j=1,2,\cdots,q$。

定义 2 设存在网格 $G=\langle V_1,V_2,E,W\rangle$ 符合定义 1，若存在 $E^*\subseteq E$，使得 E^* 中任意两边均不相邻，则称 E^* 为 G 中的一个匹配。若在 E^* 中再加上任意一条边 e，$E^*\cup\{e\}$ 中必存在着相邻的边，则称 E^* 为 G 中极大匹配。G 中边数最多的匹配称为最大匹配，其元素个数称为 G 的匹配数。若 E^* 是一个最大匹配，且 $|E^*|=\min\{|V_1|,|V_2|\}$，则称 M 为 G 中的一个完备匹配。

在上述定义的基础上，根据 Hall 定理，可得如下定理 1。

定理 1 设存在网格 $G=\langle V_1,V_2,E,W\rangle$ 符合定义 1 和定义 2，则 G 中存在从作业 V_1 到服务 V_2 的完备匹配的充分必要条件是当且仅当 V_1 中任意 k 个顶点至少与 V_2 中 k 个顶点相邻，$k=1,2,\cdots,|V_1|$，$|V_1|\leqslant|V_2|$。

为了便于用计算机程序实现完备匹配的判断，给出如下定理 2。

定理 2 设存在网格 $G=\langle V_1,V_2,E,W\rangle$ 符合定义 1 和定义 2，则 G 中存在从作业 V_1 到服务 V_2 的完备匹配的充分必要条件是：

1)V_1 中每个顶点至少关联 t 条边，其中 $t\geqslant1$；

2)V_2 中每个顶点至多关联 t 条边。

称定理 2 中的条件为 t 条件。

3 网格作业-网格服务匹配问题及其最优化解

在网格环境下，为了保证电子商务或电子政务等核心多媒体应用的服务质量(QoS)，采用的典型策略是引入预留机制。引入预留使得用户有可能知道所需要的服务资源的能力，比如网络带宽、CPU 时钟周期和硬盘空间容量等，但是可能不知道这些资源将执行什么应用。事实上，用户只希望其应用在所给的资源等级上可保证任务的执行而不关心也不了解执行任务所需要的具体资源性能，另一方面，

资源提供者也不希望公开其资源能力将如何使用的细节信息而通常只承诺用户可获得的对资源能力的保证。因此从控制原理的角度分析，包含作业和Jobmanager、服务和ServiceManager的网格可被看作一个黑箱或灰箱。从控制系统的性能指标看，一般而言，服务匹配方案的主要目的是负载平衡，但对于预留的服务而言，除了要保证各个服务分担负载，还必须考虑用户的满意程度，避免某些用户的作业执行时间过长。

为了既满足负载平衡，又保证作业执行时间，MatchMaker必须在Sensor所检测到的有关作业和服务的运行状态的反馈信息如作业完成时间 w_{ij} 基础上，求解出作业和服务之间的完备匹配。

下面给出符合定义1和定义2的满足t条件的网格作业-网格服务匹配的优化问题的数学形式：

求匹配 $e_{ij}=e_{ij}^*$，使得 $\min\sum_{i=1}^{t}\sum_{j=1}^{t}w_{ij}e_{ij}$

其中 $w_{ij}\in W$ 表示在服务 j 上完成作业 i 所花费的时间。e_{ij} 的取值范围为1或0。

$$e_{ij}=\begin{cases}1 & \text{指派作业 } i \text{ 到服务 } j \text{ 上运行时}\\ 0 & \text{不指派作业 } i \text{ 到服务 } j \text{ 上运行时}\end{cases}$$

$$\text{约束条件：}\begin{cases}\sum_{i=1}^{t}e_{ij}=1,\ j=1,2,\cdots,\ t\\ \sum_{j=1}^{t}e_{ij}=1,\ i=1,2,\ \cdots,\ t\end{cases}$$

当 i,j 的维数比较少时，可以直接用图形方法求解。当 i,j 的维数比较多时，显然可以采用矩阵表达集合 W 和 E 的方式进行求解。

该最优化问题求解是根据如下的性质：如果从权矩阵 $W=\{w_{ij}\}$ 的一行(列)各元素分别减去该行(列)的最小元素，得到新矩阵 $W'=\{w'_{ij}\}$，那么，以 W' 为系数矩阵的匹配问题的最优化解 $E^*=\{e_{ij}^*\}$ 和原问题的最优解相同[10,11]。

该最优化问题求解算法如下：

1)置匹配矩阵 E 的各个元素的值为0。

2)从权矩阵的每行元素减去各行的最小元素，如果某列存在全部非零元素时，再从列减去最小元素，使得各行各列至少有一个为零元素。

3)由新的权矩阵 W' 中零元素最少的行开始，选出一处零元素，即选择 i 和 j，使 $w'_{ij}=0$。

4)令选出的零元素 w'_{ij} 对应的匹配矩阵的元素 $e_{ij}=1$。

5)令选出的零元素 w'_{ij} 所在的权矩阵 W' 中第 i 行，第 j 列的其它零元素取某个极大值，如可取 w'_{ij} 中最大元素的值或取计算精度允许的最大值。

6)如果匹配矩阵 E 中每行每列都有且仅有1个元素的值为1，则求解完成，并记 $E=E^*$，否则返回第3步。

上述算法中 E 的维数是 t，符合完备匹配的 t 条件.因此该算法一定收敛。

4 计算机仿真和结果分析

为了在网格系统上实现本文给出的优化算法，采用MATLAB COM Builder将程序编译为单独的ActiveX对象，并由VB.Net语言编写程序调用该对象。仿真数据为，已知任务可分解为并发作业A，B,C,D,它们分别在集群服务甲、乙、丙、丁上运行时的完成时间权矩阵W如下：

$$W=\begin{pmatrix}30 & 100 & 90 & 40\\ 120 & 40 & 160 & 130\\ 80 & 120 & 140 & 130\\ 50 & 80 & 100 & 70\end{pmatrix}$$

根据优化算法求得匹配矩阵 E^* 如下：

$$E^*=\begin{pmatrix}0 & 0 & 0 & 1\\ 0 & 1 & 0 & 0\\ 1 & 0 & 0 & 0\\ 0 & 0 & 1 & 0\end{pmatrix}$$

它表示由甲完成D任务，乙完成B任务，丙完成A任务，丁完成C任务。此时总完成时间为：

$$\min\sum_{i=1}^{t}\sum_{j=1}^{t}w_{ij}e_{ij}=w_{14}+w_{22}+w_{31}+w_{43}$$
$$=40+40+80+100=260\text{ms}$$

多个作业并发运行时整个任务的完成时间为完成丁作业的时间100ms。若完成任务的费用与任务完成时间存在函数关系时，则上述优化可适用于降低请求网格资源的经济成本。

事实上，在本实例中存在最多 $C_4^1C_3^1C_2^1C_1^1=24$ 种不同的匹配策略，图-1是采用穷举搜索法时不同匹配策略下网格作业的总完成时间分布曲线：

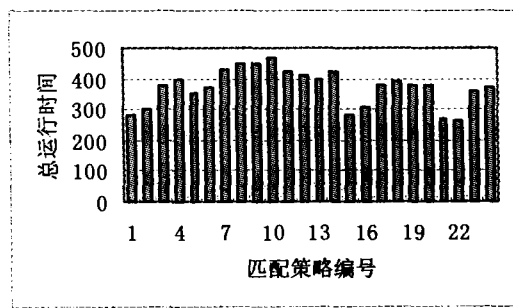

图1 不同匹配策略运行结果分析

由图1可见，最优策略的编号为22，其总运行时间为260ms，与基于权矩阵的服务匹配优化算法的结果相同；而最差策略的编号是10，其总运行时间为470ms，不同的服务匹配策略之间的总运行时间最大相差210ms。在穷举算法下各种策略的平均总运行时间为371.25ms，本文给出的优化算法与之相比使服务匹配系统性能改善42.78%。

当然，当网格作业和网格服务的数量增加时，穷举算法是低效的，而且可能产生组合爆炸。因此，我们考虑比较常用的搜索策略——爬山法相比较，并以运行时间为爬山法的启发条件，计算从 $W(1,1)$

开始，此时运行时间为 30ms，此后以运行时间增长最小为目标进行搜索，计算结果如图 2 所示。此时的最优策略是甲完成 A 作业，乙完成 B 作业，丙完成 C 作业，丁完成 D 作业，总运行时间为 280ms。与本文给出的优化算法相比，爬山法给出的是次优解，而最优解与之相比使服务匹配系统性能改善 7.69%。

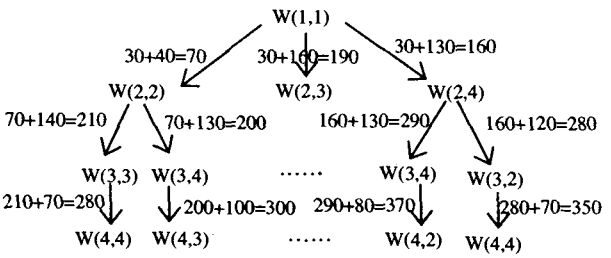

图 2　爬山法的搜索结果

结论　本文通过在网格服务模型中引入代表作业运行时间或运行成本的权函数，给出了网格作业-网格服务匹配的优化解。网格服务管理中的服务匹配问题是个十分复杂的问题，本文的方法在理论上给出了全局最优解，适合用于电子商务和电子政务核心多媒体应用中保证服务质量的预留资源的分配。但是该方法对于系统的动态性考虑仍不足，仍然有缺陷，这是将来进一步的工作。

参 考 文 献

1　Foster I, et al. The Grid: Blueprint for a New Computing Infrastructure(2nd Edition). Morgan Kaufmann, 2004

2　Czazowski K, Foster I, et al. A resource management architecture for metacomputing systems. In : 4th workshop on Job Scheduling Strategies for Parallel Processing , Springer-Verlag, Heidelberg, 1998. 62～82

3　Czazowski K, Foster I, et al. Co-allocation services for computational Grids. In: 8th IEEE International Symposium on High Performance Distributed Computing, IEEE Computer Society Press, Los Alamitos, CA, 1999

4　Raman R, Livny M, et al. Matchmaking: distributed resource management for high throughput computing. In: 7th IEEE International Symposium on High Performance Distributed Computing, IEEE Computer Society Press, Los Alamitos, CA, 1998

5　Foster I, Kesselman C, Nick J, et al. Grid services for distributed system integration. IEEE Computer, 2002, 35(6): 37～46

6　单志广，戴琼海，林闯，等. Web 请求分配和选择的综合方案与性能分析. 软件学报，2001，12(3)：355～36

7　曲绍刚，杨广文，林 闯，史树明. 基于完成时间的任务分配方案与性能分析. 计算机研究与发展，2005，42(8)：1397～1402

8　Tierney B, Johnstone W, et al. The NetLogger methodology for high performance distributed systems performance analysis. In: 7th IEEE International Symposium on High Performance Distributed Computing, IEEE Computer Society Press, Los Alamitos, CA, 1998

9　The Open Group. Systems Management: Application Response Measurement: [Technical Standard C807]. Available at: www.opengroup.org/publications/catalog/c807.htm

10　杨扬. 离散事件系统理论与柔性制造系统. 冶金工业出版社，1994. 219～221

11　殷剑宏，吴开亚. 图论及其算法. 中国科学技术大学出版社，2003. 192～199

(上接第 10 页)

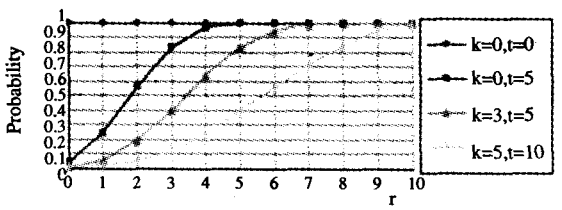

图 3　相邻概率 P 与有效半径 r 的关系（k、t 不变时）

(1)从图 3 中可以看出，在初始状态 k=0、运动时刻 t=0 时，两个节点的相邻概率 P 不随有效半径 r 变化，均为 1；其他的情况下，P 随着有效半径 r 的增大而增大。

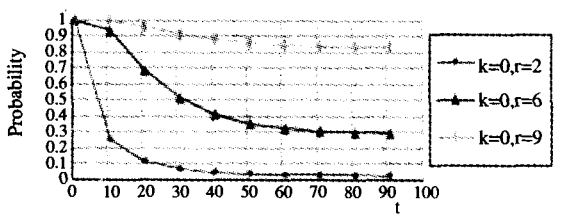

图 4　相邻概率 P 与系统时间 t 的关系(k、r 不变时)

(2)如图 4 所示，在初始状态 k 和时刻 r 一定时，两个节点的相邻概率 P 随系统时间 t 的增大而减少。

(3)从图 5 中可以看出，在初始状态 r 和时刻 t 一定、初始状态 k 不同时，两个节点的相邻概率 P 随着初始状态 k 的增大而减少。

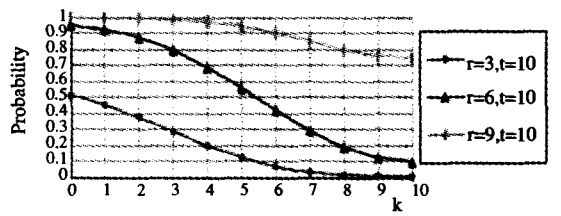

图 5　相邻概率 P 与系统的初始状态 k 的关系(r、t 不变时)

参 考 文 献

1　Royer E M, Toh C K. A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks. IEEE Personal Communications, 1999, 6(2): 46～55

2　Broch J, Maltz D A, Johnson D B, et al. A Performance Comparison of MultiHop Wireless Ad Hoc Network Routing Protocols. In: Proceedings of ACM/IEEE MobiCom'98, 1998. 85～97

3　Johnson D B, Maltz D A. Dynamic Source Routing Protocol for Mobile Ad Hoc Networks. IETF Internet Draft, 1998

4　方建超，王汉兴，贾维嘉. ad hoc 网络的离散时间马氏链建模及分析. 计算机工程，2004，30(5)：98～101

5　Markoulidakis J G, Lyberopoulos G L, Tsirkas D F, et al. Mobility modelling in third generation mobile telecommunication systems[J]. IEEE Personal Communications, August, 1997, 4(3): 41～56

6　Http://www.cs.bham.ac.uk/～dxp/prism

7　Jacquet P, Laouiti A. Analysis of Mobile Ad Hoc Network Routing Protocols in Random Graph Models: [Institut National de Recherche en Informatique et en automatique. Rapport de Recherche n°3835]. 1999