

网格服务可管理性模型及策略研究

王元卓^{1),2)} 林 闯¹⁾ 杨 扬²⁾ 单志广³⁾

¹⁾(清华大学计算机科学与技术系 北京 100084)

²⁾(北京科技大学信息工程学院 北京 100083)

³⁾(国家信息中心信息化研究部 北京 100045)

摘 要 随着网格技术和应用的不断发展,一些关键领域或业务应用场景要求网格系统提供更好的服务质量(QoS),而具有较高可管理性的服务管理策略可以在提供较高服务质量的同时减小系统的开销.文中提出了网格服务的干扰和可管理性的形式化描述及量化计算方法,并建立了描述干扰发生过程的网格系统随机 Petri 网(SPN)模型.之后文章对服务管理策略进行了分类研究,并根据 SPN 模型的仿真计算结果,对各类策略对网格服务的干扰情况的影响以及对网格服务可管理性的量化评价结果进行了比较和分析,为具有高可管理性的策略的设计提供依据.

关键词 网格服务;可管理性;干扰;服务管理;随机 Petri 网

中图法分类号 TP311

Research on Manageability of Grid Service Model Method and Management Strategies

WANG Yuan-Zhuo^{1),2)} LIN Chuang¹⁾ YANG Yang²⁾ SHAN Zhi-Guang³⁾

¹⁾(Department of Computer Science and Technology, Tsinghua University, Beijing 100084)

²⁾(Information Engineering School, University of Science and Technology Beijing, Beijing 100083)

³⁾(Department of Information Research, State Information Center, Beijing 100045)

Abstract With the development of Grid technology and Grid application, higher requirements are imposed on the efficiency and quality of the grid system. And the service management is a necessary method to improve quality of service (QoS). In this paper, the definition and formalized description of Grid service churn and manageability are proposed. And a stochastic Petri net (SPN) model is upbuilt to describe the process of Grid service with churns. On the other hand, service management strategies are classified and researched. According to the computing results of SPN model, the influences of these strategies to churn and the quantifying results of manageability are analyzed. This paper can provide the reference to actual application.

Keywords grid service; manageability; churn; service management; stochastic Petri net (SPN)

1 引 言

网格计算^[1]现已不再仅仅局限于科学研究,工

业界与学术界联盟正致力于使网格计算在更广泛的领域得到推广和应用.网格计算的一个重要特征就是在互联网环境下的资源共享与协同问题求解,十年来经历了从面向计算到计算与服务相融合的发展

收稿日期:2007-06-25;最终修改稿收到日期:2007-11-23.本课题得到国家“九七三”重点基础研究发展规划项目基金(2003CB314804)、国家自然科学基金(90718040,60673187,60673054,60673160,60803123)资助.王元卓,男,1978年生,博士,助理研究员,主要研究方向为网络安全分析、系统性能评价、网格计算等. E-mail: yzwang@csnet1.cs.tsinghua.edu.cn.林 闯,男,1948年生,博士,教授,博士生导师,主要研究领域为计算机网络、系统性能评价、随机 Petri 网、可信网络与可信计算.杨 扬,男,1955年生,博士,教授,博士生导师,主要研究领域为多媒体通信、网格计算等.单志广,男,博士,研究员,研究领域为计算机网络、Petri 网理论与应用等.

过程,面向服务的开放网格体系结构已成为网格技术发展的重要趋势。通过网格服务^[2],网格系统可以完成各种的用户请求,随着网格技术和应用的不断发展,一些关键领域或业务应用场景,如实时数据计算及传输、多媒体展现等都对网格的服务质量(Quality of Service, QoS)^[3]提出了更高的要求,有效的服务管理是提高网格服务质量的必要手段。但是,由于网格资源的动态性和自治性,网格节点会动态地加入或撤出网格系统;同时,网格资源的异构性和广域分布等特性,使得网络传输过程中以及任务执行过程中网格节点或子节点都有可能受到故障或操作事故等意外的影响,我们把这些影响称作对网格服务可管理性的干扰^[4],它的存在导致了整个网格系统的管理复杂度的增加。因此,有必要对这个问题进行深入的研究,而目前对网格服务可管理性的研究较少,且大多集中在定性上,缺乏对网格服务管理策略进行定量分析的有效理论和方法。同时,在管理策略的设计中也缺乏对可管理性因素的考虑,使得虽然在某些策略作用下,网格系统能够提供一定质量的服务,但却可能花费较大的代价。

随机 Petri 网(Stochastic Petri Nets, SPN)^[5]是图形的、数学的模型和分析工具,它已广泛地应用于计算机科学、网络管理与控制建模以及系统性能评价等领域之中,并已充分显示了它的有效性,它的并发、资源共享描述特性很适合于网格系统建模和资源管理策略的研究^[6]。

针对上述问题,在前期工作的基础上,我们提出网格服务可管理性的概念和形式化描述方法,并通过建立基于随机 Petri 网的网格服务模型,对网格服务管理策略进行量化分析比较,研究如何在任务分配前以及干扰发生后,通过选择合适的节点,来提高网格服务的可管理性。

本文第 2 节介绍相关工作,同时进一步阐明本文与相关工作的差异和深入研究的意义;第 3 节提出网格服务可管理性的概念及其形式化描述方法;第 4 节研究网格服务的随机 Petri 网模型,为管理策略的量化分析以及网格服务可管理性的计算提供基础;第 5 节对各种管理策略的特点进行分类研究,并进行 SPN 模型的仿真计算;第 6 节分析各种管理策略在不同的环境和条件下的特点及其对网格服务可管理性的影响;第 7 节总结全文,并展望下一步的研究工作。

2 相关工作

目前,已有的管理策略大都是基于传统的分布式系统提出的。文献[7]在多管理域的大规模网络计算系统中引入了 QoS 驱动的管理策略,利用效用函数量化服务质量等级,提出作业级动态调度算法来优化性能和完成时间。文献[8]提出了一种基于网格服务 workflow 模型的容错管理策略,该策略在任务级和 workflow 级上对应用错误进行处理,但其对错误的检测是基于静态心跳机制的,不能较好地满足网格系统的动态性,且可扩展性较差;文献[9]通过改进传统的 Min-Min 调度启发式,提出了一个自适应的、性能驱动的 Min-Min 启发式算法。同样,文献[10]以平均响应时间作为性能要求,提出以 Sufferage 为基算法的算法,此算法性能在期望完成时间和平均响应时间方面显著优于 Max-Min 等算法。文献[11]提出的策略中通过对服务失效数据的分析,来选择一个性能最佳的计算节点来执行服务。这些服务管理策略大多以系统性能或服务质量为设计目标,在一定程度上忽略服务本身的可管理程度。文献[12]提出了管理质量的概念,认为服务的可管理程度应该和服务质量一样是可以度量的。文献[13-14]以及 Potts 等^①从不同角度对服务可管理性进行了描述,但都缺乏针对网格特点的服务管理模型。本文将重点研究网格环境下服务可管理性的形式化定义,并通过描述干扰事件发生过程的网格系统 SPN 模型,研究各种典型服务管理策略的可管理性的情况。

3 网格服务可管理性

为了探讨网格服务可管理性的问题,必须首先对可管理性的研究范畴进行界定,进而研究影响网格服务可管理性的因素。现有文献对可管理性并没有给出精确的定义,但是这个术语经常作为系统设计思想。研究网格服务的可管理性,首先要研究网格服务状态和网格资源的特性。在不同的环境下网格服务状态繁多,对可管理性的研究,就是从管理角度来描述服务操作的可管理程度。

网格的功能要通过网格服务的形式来具体实现,而具体的操作就是网格系统与其外部环境间的

① Mark Potts, Igor Sedukhin, Heather Kreger. Web Services Manageability — Concepts. <http://www.ibm.com/developerworks/webservices/library/ws-manage/>

作用,它可以由输入、输出向量来描述,输入、输出向量则由一定的信息流量来表示. 网格资源管理系统从其环境中所获得的信息流量的总和,构成该系统的输入向量;网格服务的结果又体现为转向其环境的网格信息流量,这些流量的总和构成该系统的输出向量. 在完全可管理的网格系统中,确定的输入向量一定对应着确定的输出向量. 但由于现实环境中存在网格节点的动态性以及开放的网格系统更易受到故障、事故以及安全事件等的内外因素的干扰,故通常的网格服务的结果都会带有一定的不确定性. 通过上面的分析,我们给出网格服务可管理性的定义如下.

定义 1. 网格服务可管理性. 它是指在网格节点受到内外干扰的情况下,网格系统为用户提供服务的过程及结果可以预期的程度.

分析网格服务的可管理性,并进行量化评估,首先,需要对整个网格服务状态的变化情况进行形式化的描述.

假设在任意时刻,网格系统中有 n 个节点,且每一个节点存在有效和失效两种状态,其中那些处于有效状态的节点既可能是正在被使用,也可能是处于可用状态等待被使用. 节点按照某种过程失效和恢复. 我们称一个节点处于有效状态的平均时间为该节点的服务周期. 在任意时刻,网格资源管理系统可以选择添加节点或者从正在使用的节点中删除节点,使节点从有效状态变成使用或者退出的状态. 处于使用状态的节点目标数量满足 $k = \alpha n$, $0 < \alpha \leq 1$, 其中 n 表示网格节点的数量,则除非有少于 k 的节点处于正常状态,我们考虑的替换策略将是完全匹配的.

参考文献[15],我们对网格资源和网格服务进行下列数学描述:若当前网格系统有 n 个可用的节点,每一个节点可完成 x 种网格操作,则节点 j ($j = 1, 2, \dots, n$) 在完成这些网格操作时的能力可描述为 $\{N_{1j}, N_{2j}, \dots, N_{xj}\}$. 假设通过这些操作能够实现满足用户需求的 k 类网格服务,可以表示为 $\{P_{1j}, P_{2j}, \dots, P_{xj}\}$. 前面我们分析了,网格节点在提供服务时可能受到各种意外干扰的影响,在服务执行过程中,每个节点的服务能力都会受到这些干扰的影响.

下面来讨论服务状态的变化,我们认为一个网格服务在其执行节点操作能力降低后,将会导致服务状态的迁移. 假设 S_m 是网格服务的第 m 个状态,则此状态下第 j 个网格节点的正常服务能力

可描述为 $N_j^m = \{N_{1j}^m, N_{2j}^m, \dots, N_{kj}^m\}$, 其中界限为 $[\underline{N}_{ij}^m, \overline{N}_{ij}^m]$, 代表第 i 个操作的下界和上界. 同时,在状态 S_m 下,节点 j 执行的 k 类网格服务可以表示为 $P_j^m = \{P_{1j}^m, P_{2j}^m, \dots, P_{kj}^m\}$, 其中每一个成员 P_{ij}^m 值的边界由 $[\underline{P}_{ij}^m, \overline{P}_{ij}^m]$ 决定,代表了完成第 i 类服务所需资源的下界和上界.

假设当前网格服务状态为 S_a , 随着一些干扰事件的发生,如果节点 j 的 i 操作 N_{ij}^a 没有超过 $[\underline{N}_{ij}^a, \overline{N}_{ij}^a]$ 的范围,即该操作可以在此节点上正常执行,并且该操作参数在服务 l 的限制范围 $[\underline{P}_{lj}^a, \overline{P}_{lj}^a]$ 内,即满足该服务的要求,则网格服务可在当前状态下继续执行. 如果干扰事件导致一个或多个参数超过当前状态的范围,我们认为网格服务将会发生状态的变化. 可以看到干扰事件的发生使得网格服务状态发生了改变.

因此,网格服务状态 S_m 的变化,受到二元组 (N^m, P^m) 的影响,简单来说,可以将操作空间和服务空间分别划分为 3 个区域,如图 1 所示.

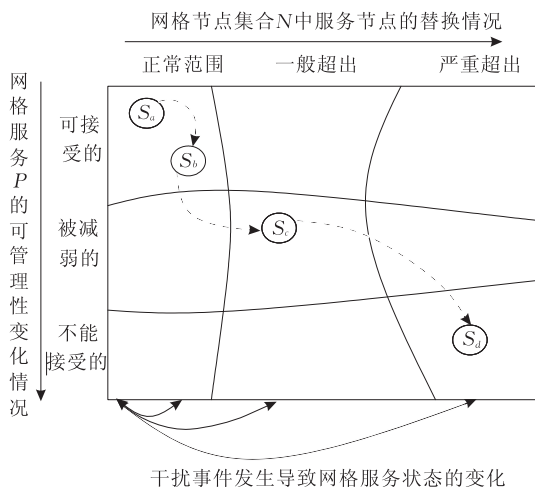


图 1 干扰发生导致网格服务状态变化的情况

根据前面的分析网格服务可管理性的形式化描述可以表示为

$$M(S_m) = 1 - \sigma_m \times \left(\frac{C}{C+1} \right)^\mu \quad (1)$$

其中, S_m 为服务当前所处的状态, σ_m 表示在状态 S_m 下干扰对可管理性的影响程度, C 表示系统发生的干扰率, μ 是干扰的调节系数,对不同类型的干扰可设置不同的调节系数. 可见要研究可管理性,首先要研究系统干扰发生的情况.

下面我们以系统中节点被替换的速率来给出网格服务中干扰的定义和形式化描述.

定义 2. 干扰. 给定一个处于使用状态的节点

发生改变的序列, 设 U_i 为第 i 个节点发生变化后, 处于使用状态的节点的集合, 其中 U_0 为初始集合. 当某些意外事件发生时, 系统的状态将会发生改变, 而干扰就是系统发生意外事件的总数, 可以通过运行时间 T , 将干扰形式化地描述成

$$C = \frac{1}{T} \sum_{eventsi} \frac{|U_{i-1} \ominus U_i|}{\max\{|U_{i-1}|, |U_i|\}} \quad (2)$$

其中, \ominus 是取差异的符号, 我们分别记录一个节点失效事件和网格资源管理系统对失效事件的响应时间. 因此在运行 T 时间内, 如果我们开始有 k 个节点处在使用状态, 其中 2 个节点先后失效, 并且网格资源管理系统又重新分配了两个可用状态的节点, 则此时的干扰为 $\frac{1}{T} \left(\frac{2}{k} + \frac{2}{k} \right)$. 如果 k 个被使用的节点每一个都失效了, 并且都没有可以替换的节点, 则干扰可表示为 $\frac{1}{T} \left(\frac{1}{k} + \frac{1}{k-1} + \dots + \frac{1}{1} \right) \approx \frac{1}{T} \ln k$.

这里一个重要的假设就是在这个定义中, 我们认为在出现节点失效后将优先使用那些没有被使用过的节点, 而不是那些失效的并且被替换的节点. 在实际有状态的系统中, 这样的假设是合理的^[16].

除了上述我们研究的节点服务失效所产生的干扰外, 为了提高系统的性能, 有时网格系统还需要采用一些动态策略, 来动态调整任务完成的过程. 而这样的过程也会产生对网格服务的干扰.

定义 3. 服务迁移. 当网格系统中存在可用节点时, 为了提高系统的性能, 有时系统会根据某些动态节点选择策略将一个节点上正在执行的服务迁移到另一个节点上执行, 即将一个节点由正在使用状态变为可用状态.

服务迁移的前提是系统中存在可用的节点, 假设某一时刻, 系统中有 k 个节点在使用, 且存在其它可用的节点, 我们来考虑下面的过程, 根据前面的定义, 每一次服务失效或服务迁移的代价是 $1/k$, 并且我们能够这样来描述一个策略 s , 使用中的 k 个节点可以表示为 s^1, \dots, s^k , 其中每一个 $s^i = (s_i^1, \dots, s_i^j)$ 可表示一个使用序列, s_i^j 被选择来执行 s_{i-1}^j 上的服务. 主动的服务迁移可以提高服务质量或系统的吞吐量, 但同时也会对服务的运行造成干扰, 对服务迁移所带来的干扰情况, 我们给出下面的定理, 并参考文献^[15]给出证明.

定理 1. 对于确定的失效模式, 如果根据某个动态策略 V , 使得一个网格节点 v_j^i 发生了服务迁移, 此时产生干扰 C , 那么对另一个不发生服务迁移的

策略 W 产生的干扰 $C' \leq C$.

证明. 令 v_j^i 是一个在下一个失效时刻 t_2 之前的某一个时刻 t_1 发生服务迁移的节点. 我们来分两种情况进行证明:

(1) 节点 v_j^i 在 $[t_1, t_2)$ 内没有被使用. 让 $v_l^i (l > j)$ 是采用策略 V 时, 在 t_2 时刻使用序列中的节点. 策略 W 通过删除节点使用序列 i 中 j 到 l 之间的所有节点, 在时间 $[t_1, t_2)$ 内只保留节点 v_j^i . 这两个策略只在 $[t_1, t_2)$ 时间内有所不同, 则策略 V 中至少要产生一次迁移的干扰, 而此时策略 W 中最多只发生一次失效.

(2) 节点 v_j^i 在时间 $[t_1, t_2)$ 内一次被使用. 如果在节点 v_j^i 使用序列 i 中再次被使用, 则我们只需要从策略 W 的节点使用序列中删除受到干扰的节点. 否则, 如果它被其它节点序列 l 所使用, 则我们变换节点使用序列如下: 序列 i 保留 v_j^i 直到它失效, 然后从时间 t_2 往前参照序列 l 的选择. 序列 l 根据序列 i 前面的选择进行选择. 很明显, 如此构建策略 W 只能够减少失效和再次选择总的次数. 重复上述过程, 定理得证. 证毕.

服务迁移的发生虽然对服务过程带来了一定的干扰, 但由于同时也提高了服务的性能, 从而影响着服务状态的变化. 即虽然使得式(1)中的 C 有所增长, 但同时减少了 σ 对可管理性的影响.

4 网格服务模型

目前, 尚没有针对网格服务可管理性的模型和评价理论, 对网格服务特性的定量分析需要建立相应的数学模型. 这里的数学模型可以是运用运筹学、随机理论等数学的方法和技术建立的模型, 主要用于定量地描述网格系统的有关动态和静态的过程, 对网格服务进行定量的分析和数值计算.

具体可分为确定模型和随机模型等. 排队模型是一种确定性的数学模型, 它是很好的网络建模工具, 其确定性表现在: 队列根据到达先后确定排队先后; 服务器的服务策略是预先确定的; 信源分布也是事先确定的. 这里的服务器可以用来描述网格系统中的某个独立功能的节点, 如终端、链路或网格节点, 也可以是某层的网络协议, 而队列刻画的是网络系统中待处理的对象的序列关系. 随机模型是以随机过程为基础, 主要包括马尔可夫回报模型、随机进程代数以及随机 Petri 网等, 这些基于状态的随机方法更容易对网格系统状态进行全面有效的描述,

精确刻画网格系统随机行为以及节点之间的相互关系,便于计算各种分析指标. 网络事件的发生时间一般可设为随机变量,网络服务行为具有随机性的特点;同时,网络节点间的相互作用又具有很强的相关性,系统行为具有既随机又相关的特点,对网络服务可管理性的建模和分析要综合考这些特性. 因此,在研究网络服务的可管理性模型方法时还应该考虑网络系统的结构模型,既要刻画网络系统的物理实体和逻辑实体的操作,还要刻画网络的具体结构以及节点之间的关系.

研究可管理性,要充分研究各种方法的特点以及分析网络服务可管理性评价的需要. 随机 Petri 网模型既有马尔可夫过程同构的特性,便于数学求解,又有图形化描述系统结构的特点. 而直接使用马尔可夫模型,却很难对马尔可夫模型进行扩展. 例如在实际系统中,一旦系统的资源增加或减少,相应的马尔可夫链的结构就可能会发生很大的变化,而在随机 Petri 网模型中,只要通过增加相应位置中的标记数就可以建立相应的模型. 并且随机 Petri 网可以在一个系统模型的框架上采用图形化的方式完成系统的描述、分析以及系统的验证和测试,这是其它的方法所不具备的功能.

我们既要考虑建模的精确性,又要考虑求解的可行性,研究网络服务可管理性模型与管理策略的关系分析与求解,研究各种模型和方案的复杂性与有效求解之间的折衷. 由于网络系统庞大而复杂,建立整个系统的可管理性模型一般比较困难,而且对于状态空间庞大的随机模型的求解较为困难,故可以从局部出发建立局部的可管理性模型,再结合其它方法求得网络可管理性的次优解. 随机 Petri 网理论在很多方面的发展已很完备,它已有很多成熟的、有效的软件工具,可大大简化使用者进行建模和求解的过程. 如 ESPN, GreatSPN, SPNP, URF-2, TOMSPIN, UltraSAN 等,因此,它是大型复杂网络系统模型分析的有力工具^[17].

下面,我们用 SPN 建立带有干扰事件的网格系统模型,如图 2 所示. 模型中,长方型表示时间变迁(timed transition),黑色的线代表顺时变迁(immediate transition). 任务的到达、被执行、用户服务请求的到达以及对干扰发生过程的描述都用时间变迁来表示,且假设这些变迁的响应时间服从指数分布. 任务被分配到网络节点以及网络服务被干扰事件影响等情况我们用顺时变迁表示. 圆圈表示位置(place),黑点表示位置中的标识(token). 在

模型中标识可以表示来自用户的服务请求,也可以表示网络资源. 用位置表示任务等待队列,并用这些表示队列的位置中的标识的数量来表示等待队列中等待任务的数量. 在此模型中,任务符号的下标表示任务的类型,并且另一个下标表示相应的网络节点.

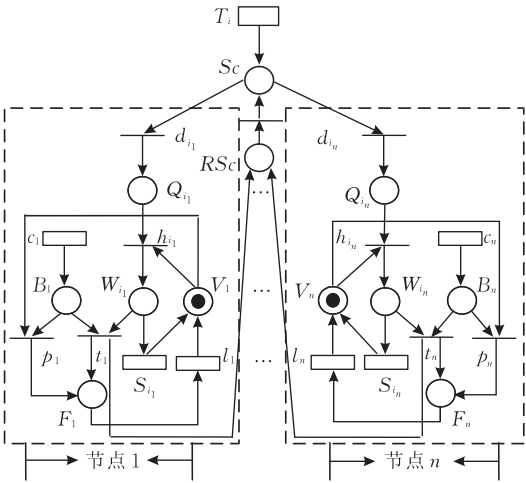


图 2 带有干扰事件的网格系统 SPN 模型

图 2 中变迁和位置的符号描述如下,其中 $(1 \leq i \leq m, 1 \leq j \leq n)$:

(1) 变迁:

T_i 表示任务 $Task_i$ 的到达,且其到达率为 λ_i .

d_{ij} 表示调度器 Sc 分配任务 $Task_i$ 到节点 j . 分配策略可以由变迁相关的谓词以及响应概率表示.

h_{ij} 表示任务 $Task_i$ 在节点 j 上执行.

s_{ij} 表示任务 $Task_i$ 正在节点 j 上执行,且平均服务速率为 μ_{ij} .

c_j 表示节点 j 受到干扰事件的影响,将会中断节点 j 正在执行的服务,干扰事件发生的速率为 η_j .

t_j 表示由于节点 j 受到干扰,正在节点 j 上运行的服务将被中断,并且任务 $Task_i$ 将被替换到新的节点上运行.

p_j 表示由于节点 j 受到干扰, V_j 中的资源将处于失效状态.

l_j 表示受到干扰的节点 j 恢复的过程,并且假设恢复的速率为 θ_j .

(2) 位置:

Sc 表示在网络资源管理系统. 根据具体的策略,启动节点 j 上的网络服务,执行任务 $Task_i$.

RSc 表示网络服务受到干扰后,任务被重新调度.

Q_i 表示节点 j 对设置的等待队列,且节点 j 等待队列的容量为 b_j .

W_{ij} 表示节点 j 正在执行网格服务.

V_j 表示节点 j 的计算资源所在的位置. V_j 中的一个标识表示一个网格服务资源.

B_j 表示节点 j 将要发生干扰事件.

F_n 表示节点 j 处于失效状态.

在上述 SPN 的模型中, 我们假设服务请求 $Task_i$ 的到达速率 λ_i 、节点 j 的平均执行速率 μ_{ij} 、节点 j 上干扰事件发生速率 η_j 都被认为是一个泊松过程^[18].

5 管理策略选择

为了更好地理解干扰事件的存在对节点选择策略的影响, 我们研究了一系列的策略, 其中既包括实际中被广泛使用的, 也包括研究领域受到关注的. 我们从不同角度来研究节点选择策略, 一方面是根据策略中是否会依据节点的信息预先判断网格节点的特点, 如平均服务时间、可用性程度等, 我们称之为可预测策略(predictive strategies). 另一方面是根据是否用一个新节点来替换一个失效节点. 其中, 固定策略(fixed strategies)不会从初始选择的集合中选择节点来替换失效的节点; 而与之相反, 替换策略(replacement strategies)是当节点失效后, 只要有可用的节点就立刻替换.

可预测的固定策略(predictive fixed strategies)可以根据节点的历史信息, 选择一些有着较好可用性的节点来长时间固定运行某些服务. 可预测的替换策略(predictive replacement strategies)可以实现动态减小干扰的影响, 而大多数启发式的策略就是根据已有信息, 选择平均服务时间最长或者服务速率最大的节点^[19-20]. 不可预测策略(agnostic strategies)经常用来描述那些无法减小干扰影响的系统. 不可预测的替换策略(agnostic replacement strategies)的最简单形式就是随机替换策略(Random Replacement, RR), 而其另一种重要的形式就是有偏好的策略(Preference List strategies, PL), 这类策略除了考虑干扰外还会根据其它方面的优化情况, 如根据某些偏好给节点排序情况, 并选择其中最可用的 k 个节点.

5.1 管理策略

(1) 可预测的固定策略

当在一个系统中配置一个服务时, 可以在系统运行前观察这些节点, 然后使用下面的任何一种启发式的策略, 针对系统的生命周期, 选择一个最好的

固定节点集合, 出现干扰时, 任务将等待该节点的恢复:

固定比例策略(fixed decent): 在观察期间运行时间最短的 50% 的节点被排除, 在保留的节点中随机选取 k 个节点(如果 $k > \frac{n}{2}$, 则选取所有保留的节点和在被排出的节点中随机选择 $k - \frac{n}{2}$ 个节点).

固定最可用策略(fixed most available): 随机选择使用时间最多的 k 个节点.

固定最大服务能力策略(fixed max ability): 选择服务能力最大的 k 个节点.

这样可以在观察期很自然地挑选出 k 个节点来使干扰最小, 但是, 不幸的是这样通常会遇到 NP 完全问题^[15]. 复杂性来源于可用节点的数量情况.

(2) 不可预测的固定策略

在这一类策略中我们只研究一个策略, 它的性能基本上与偏好优先策略是一样的.

固定随机策略(fixed random): 选择 k 个随机节点.

(3) 可预测的替换策略

下面的策略都是随机选择 k 个节点作为初始集合, 并且在使用的节点失效后, 选取一个替换的节点. 它们的不同在于所选取节点的依据.

最小期望完成时间策略(stortest expected accomplishing time): 选择从分配任务到完成服务所需时间最小的节点, 它不仅考虑队列状态而且考虑服务等待时间的情况.

最大期望策略(max expectation): 选择连续运行时间期望值最大的节点, 通过考察节点的历史服务周期, 来确定它当前的运行时间的情况.

最优替换策略(optimal): 选择下一次失效前时间最长的节点, 这个策略需要对未来情况的预判, 它可以提供一个很有用的参照.

(4) 不可预测的替换策略

随机替换策略(random replacement): 随机选取 k 个初始节点, 当一个节点失效后, 随机选取一个节点来替代它.

消极偏好排序策略(passive preference list): 给定节点的等级排序, 在一个正在使用的节点失效后, 用最优先的可用节点来替代它.

积极偏好排序策略(active preference list): 给定节点的等级排序, 在一个正在使用的节点失效后, 用最优先的可用节点来替代它. 但当那个失效的节

点恢复可用时,它比我们正在使用的节点具有更高的优先级,使用它来替换正在使用中的优先级最低的节点.

5.2 基于 SPN 模型的策略比较

根据上节介绍的不同节点选择策略,网格资源管理系统可以按照用户的具体要求,将任务分配到不同的网格节点上运行. 本节我们将在前面提出的基于随机 Petri 网的网格服务模型的基础上,比较各类型策略的特点以及在该策略下干扰对可管理性的影响情况. 下面我们分别选择固定策略中的固定最大服务能力策略、可预测的替换策略中的最小期望完成时间策略以及不可预测的替换策略中的随机替换策略和消极偏好策略,来研究它们在随机 Petri 网模型中所表现出来的不同特性. 这里我们假设在存在可用节点的时候,优先从可用节点中选择;再无可节点的时候从全部有效节点中选择. 它们在模型相应变迁中的谓词表示描述如下:

(1) 固定最大服务能力策略(FMA)

变迁 d_{i_j} 的可实施谓词 y_{i_j} 可以写作:

$$y_{i_j} : (M(Q_{i_j}) < b_{i_j}) \wedge ((\text{for } \forall k \neq j, \mu_{i_k} \geq \mu_{i_j}) \vee (\text{for } \forall k \neq j, M(Q_{i_k}) = b_{i_k})).$$

变迁 d_{i_j} 的随机开关 g_{i_j} 可以写作:

$$g_{i_j}(M) = \begin{cases} \frac{1}{\|FMA(M)\|}, & \text{如果 } j \in FMA(M) \\ 0, & \text{其它} \end{cases}$$

其中

$$FMA(M) = \{m \mid \mu_{i_m} = \max(\mu_{i_1}, \mu_{i_2}, \dots, \mu_{i_n}), \text{ 且 } M(Q_{i_m}) < b_{i_m}\}.$$

干扰发生后,中断的服务将不会被重新提交到 Sc ,而是直接进入该节点的等待队列 Q_{i_j} ,等待节点恢复服务.

(2) 最小期望完成时间策略(SEAT)

变迁 d_{i_j} 的可实施谓词 y_{i_j} 可以写作:

$$y_{i_j} : (M(Q_{i_j}) < b_{i_j}) \wedge ((\text{for } \forall m \neq j, M(Q_{i_j})/\mu_{i_j} \leq M(Q_{i_m})/\mu_{i_m}) \vee (\text{for } \forall m \neq j, M(Q_{i_m}) = b_{i_m})).$$

变迁 d_{i_j} 的随机开关 g_{i_j} 可以写作:

$$g_{i_j}(M) = \begin{cases} \frac{1}{\|SEAT(M)\|}, & \text{如果 } j \in SEAT(M) \\ 0, & \text{其它} \end{cases}$$

其中

$$SED(M) = \left\{ m \mid \frac{M(Q_{i_m})}{\mu_{i_m}} = \min \left(\frac{M(Q_{i_1})}{\mu_{i_1}}, \frac{M(Q_{i_2})}{\mu_{i_2}}, \dots, \frac{M(Q_{i_n})}{\mu_{i_n}} \right) \text{ 且 } M(Q_{i_m}) < b_{i_m} \right\}.$$

(3) 随机替换策略(RR)

变迁 d_{i_j} 的可实施谓词 y_{i_j} 可以写作:

$$y_{i_j} : M(Q_{i_j}) < b_{i_j}.$$

变迁 d_{i_j} 的随机开关 g_{i_j} 可以写作:

$$g_{i_j}(M) = \begin{cases} \frac{1}{\|RR(M)\|}, & \text{如果 } j \in RR(M) \\ 0, & \text{其它} \end{cases}$$

其中 $RR(M) = \{m \mid M(Q_{i_m}) < b_{i_m}\}.$

(4) 最小失效率偏好策略(MFPL)

在偏好策略中我们选择用各网格节点的服务能力对节点排序选择其中服务能力最好的 k 个节点作为服务节点.

变迁 d_{i_j} 的可实施谓词 y_{i_j} 可以写作:

$$y_{i_j} : (M(Q_{i_j}) < b_{i_j}) \wedge ((\text{for } \forall m \neq j, \eta_{i_j} \leq \eta_{i_m}) \vee (\text{for } \forall m \neq j, M(Q_{i_m}) = b_{i_m})).$$

变迁 d_{i_j} 的随机开关 g_{i_j} 可以写作:

$$g_{i_j}(M) = \begin{cases} \frac{1}{\|MFPL(M)\|}, & \text{如果 } j \in MFPL(M) \\ 0, & \text{其它} \end{cases}$$

其中

$$MFPL(M) = \{m \mid \eta_{i_m} = \min(\eta_{i_1}, \eta_{i_2}, \dots, \eta_{i_n}), \text{ 且 } M(Q_{i_m}) < b_{i_m}\}.$$

其中消极偏好策略是在失效后选择当时可用节点中失效率最小的节点,积极偏好策略同样先选择当时可用节点中失效率最小的节点,但当原节点恢复后,服务将被重新分配回原节点上运行.

下面,我们基于图 2 中的 SPN 模型分析上述几种策略的性能以及在这几种策略下干扰对网格服务可管理性的影响. 对具体参数的量化可以根据随机模型稳定状态的概率来进行. 我们先构建模型响应的马尔可夫链(MC). 基于马尔可夫链和状态转移概率,我们建立状态转移矩阵,并获得稳定状态的概率,从而计算各种评价参数^[21]. 我们采用 SPNP (Stochastic Petri Net Package)^[22] 软件包计算相应的参数. 在 SPNP 中,性能评价的基础是计算稳态时每个变迁的吞吐量以及每个位置的平均标识的数量.

根据前面对干扰和网格服务可管理性的形式化描述,下面我们来讨论在不同条件下相应指标的计算方法. 首先来研究干扰的计算方法.

(1) 当时间 $T < \frac{n}{\lambda}$ 时,若干扰造成的节点失效的速率小于失效节点的恢复速率,即 $\eta_j < \theta_j$,则干扰程度可以表示为

$$C = \frac{1}{T} \times \frac{2}{k} \times N \tag{3}$$

其中, T 是网格系统运行的时间, k 是当前提交的任务数, N 表示在时间 T 内干扰发生的次数, 可以通过 SPN 模型中变迁 t_i 的触发情况获得.

(2) 当时间 $\frac{n}{\lambda} \leq T$ 时, 若干扰造成的节点失效的速率小于失效节点的恢复速率, 即 $\eta_j < \theta_j$, 则干扰程度可以表示为

$$C = \frac{1}{T} \left(\frac{1}{n} \times P_0 + \frac{1}{n-1} \times P_1 + \cdots + \frac{1}{1} \times P_n \right) \times N \tag{4}$$

其中, T 是网格系统运行的时间, k 是当前提交的任务数, P_i 表示有 i 个服务同时受到干扰影响的概率, 即稳定状态下 SPN 模型中位置 RSc 中共有 i 个标记的概率.

根据图 2 中所示的 SPN 模型, 正在执行的任务由于干扰而被中断的情况可用变迁 t_j 的吞吐量总和与时间 T 之和表示. 即

$$N = TH(t) \times T = \left(\sum_{j=1}^n TH(t_j) \right) \times T \tag{5}$$

各网格系统的吞吐量应该等于所有节点吞吐量之和, 即

$$TH = \sum_{j=1}^m TH(s_{i_j}) \tag{6}$$

有 i 个服务同时受到干扰影响的概率可以表示为

$$P_i = p[M(RSc) = i] \tag{7}$$

每个等待对列中的平均长度可以表示为

$$D(Q_{i_j}) = \sum_{k=0}^b k \times P[M(Q_{i_j}) = k], \quad j = 1, 2, \cdots, n \tag{8}$$

节点 s_{i_j} 中的响应时间 RT_{i_j} 可以表示为

$$RT_{i_j} = \frac{D(Q_{i_j})}{TH(s_{i_j})}, \quad j = 1, 2, \cdots, n \tag{9}$$

网格系统对任务 $Task_i$ 的平均响应时间为

$$RT_i = \frac{\sum_{j=1}^n D(Q_{i_j})}{\sum_{j=1}^n TH(s_{i_j})}, \quad j = 1, 2, \cdots, n \tag{10}$$

下面我们使用 SPNP 软件包, 基于图 2 所示的 SPN 模型来分析各种策略对干扰和网格服务可管理性的影响. 为了避免状态空间爆炸, 并易于模型的求解与分析, 又不失一般性, 我们假设网格资源管理系统中有 3 个服务节点, 且初始时刻均处于可用状

态, 系统中的各节点具有不同的服务速率, 这些服务速率可以根据统计数据求得. 模型中各节点的参数设置如表 1 所示.

表 1 模型主要参数设置

	服务速率	干扰速率	恢复速率	等待队长
节点 1	$\mu_1 = 1$	$\eta_1 = 0.5$	$\theta_1 = 1$	$b_1 = 10$
节点 2	$\mu_2 = 3$	$\eta_2 = 1.5$	$\theta_2 = 2.5$	$b_2 = 10$
节点 3	$\mu_3 = 5$	$\eta_3 = 2.5$	$\theta_3 = 3.5$	$b_3 = 10$

下面我们来研究当任务到达速率 λ 由 0.5 变化到 5 时系统服务的变化情况. 根据 SPNP 软件计算出的参数, 采用我们前面给出的公式可以获得不同策略在干扰发生时网格系统对任务 $Task_i$ 的平均响应时间, 如图 4 所示. 同时, 我们可以设定 $\eta_i = 0 (i = 1, 2, 3)$ 来获得系统在无扰时的平均响应时间, 如图 3 所示. 按照前面给出的网格服务可管理性的定义和形式化描述, 我们选择干扰对平均响应时间的影响程度, 来确定当前网格服务可管理性所处的状态. 当干扰时的平均响应时间与正常时的平均响应时间之差 Δt 大于某个值时, 我们认为服务可管理性的状态将发生变化, 同时式(1)中的参数 σ 也将随之发生相应的改变. 这里我们假设初始时 $\sigma_i = 0$, 当 Δt 大于 1 时, σ_i 将增加 0.1, 以表示干扰对可管理性的

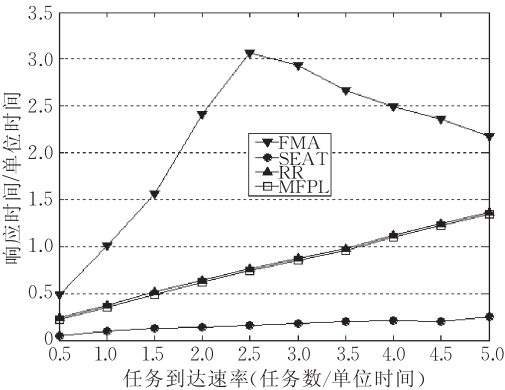


图 3 无干扰发生时网格服务响应时间

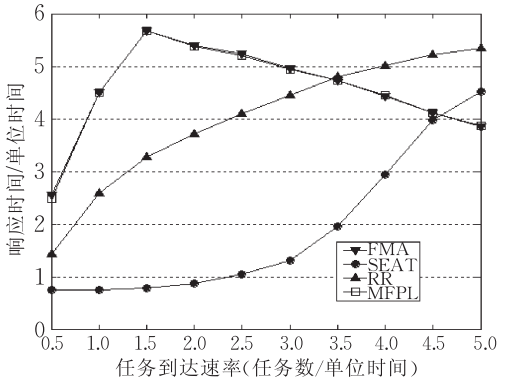


图 4 有干扰发生时网格服务响应时间

影响的增加,并取 $\mu=1$. 对各种策略下系统的干扰值和可管理性的计算结果分别如图 5、图 6 所示.

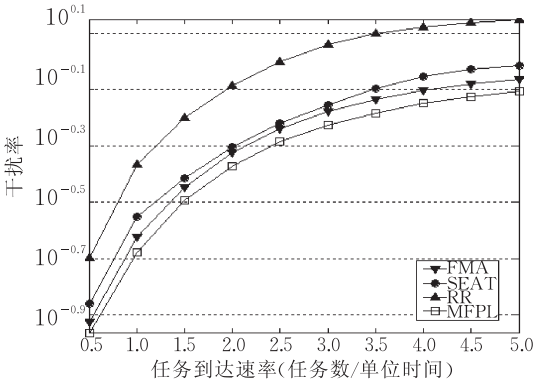


图 5 干扰随任务到达速率增加的变化情况

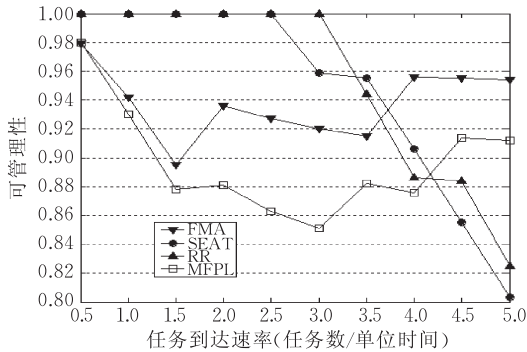


图 6 可管理性随任务到达速率增加的变化情况

6 分 析

我们可以将上述模型中模拟的过程,看作如下的更新过程.在初始时刻,网络系统中所有的节点都是有效状态.每一个节点 s_i 的平均服务周期 l_1 独立分布于其它的节点,在 l_1 时刻失效,之后要经过一个恢复时间 l_2 ,恢复到正常状态,然后再在 $2l_1 + l_2$ 时刻失效,以此类推,直到运行到某个给定时间 T ,在此过程中网格节点将可能分别处于可用状态、失效状态和正在被使用状态.当任务到达速率较小时,系统将可以保持部分节点处于可用状态,而当任务的到达速率较高时,系统中的有限节点将都被任务所占用,即此时系统中的节点将最多只有失效和被使用两种状态.此时新提交的任务将根据具体策略被分配到相应节点的等待队列之中.而当任务到达速率达到一定值后,将超过服务能力和队列的容量,此时,新提交任务将被丢弃.我们感兴趣的是当 $T \rightarrow \infty$ 时干扰和可管理性的期望值.这里我们认为 $l_1 \gg l_2$,并且系统的平均失效速率为 η .

6.1 固定策略和偏好策略

根据前面对各种策略的描述,我们知道,当系统中还存在可用节点时,管理策略将在这些可用节点中选择最合适节点执行任务,而当系统中所有节点都不处于可用状态时,管理策略将在所有有效节点(除失效节点外的节点)中选择最合适的节点,并将任务分配到相应节点等待队列中.故从图 3 中我们可以看到在任务到达速率较小时,固定策略的选择只是局部最优,所以系统的平均响应时间相对较高,但随着任务到达速率的不断增加,系统开始从全局选择节点,故在无干扰时平均响应时间在速率达到 2.5 后开始下降,而当干扰存在时,在同等任务强度下,系统将更早达到无可节点状态,如图 4 中所示.根据前面的假设和我们对干扰的定义,固定策略下的干扰可以简化为 $2/kT$ 失效的次数,即每次失效是 $1/k$,每次恢复也是 $1/k$,经过时间是 T .当 $T \rightarrow \infty$ 时,系统达到稳定状态,任何一个节点失效的次数都逼近它的期望值 T/η ,因此,在选择 k 个节点执行服务时,总的失效数量可近似为 Tk/η .因此,所有固定策略期望干扰的结果为 $(2/kT) \times (Tk/\eta) = 2/\eta$.

对于选取节点失效率作为偏好的策略 MFPL,其在无干扰发生时的情况将近似于随机策略 RR,如图 3 所示.而在有干扰发生时,在于选取的偏好,即各节点的失效率差别较为明显时,其平均响应时间近似于固定策略.下面我们来具体分析消极偏好策略下的干扰情况,根据前面的定义,策略将首先按照偏好为所有可用的网格节点排序.像固定策略一样,一个失效节点 s 导致我们将要为失效和替换耗费 $2/kT$.因为 $l_1 \gg l_2$,下一次某个其它节点失效时, s 一定会被替换,也就是说,在任何时间,集合排序列表中最多只有一个不能使用的节点.随着任务到达速率的增加,使用节点的失效频率增加,因此我们重新分配到 s 的时间越来越短.特别是在下一次失效前,我们重新分配到 s 的概率接近 1.因此,对于较大的任务到达速率,消极偏好策略的干扰也接近 $2/\eta$.积极偏好策略也是相同的,只是在恢复以后,它要耗费 $1/kT$ 来重新分配到 s ,产生的干扰为 $3/\eta$.我们选取的 MFPL 为消极偏好策略,故其干扰情况与固定策略相接近.

对可管理性的计算除了受到干扰值的影响,同时还要考虑系统此时的状态变化情况,上面的仿真计算中,选取响应时间作为服务状态变化的依据,故 MFPL 策略虽然与 FMA 策略的干扰情况相近,但由于 MFPL 策略的响应时间受干扰影响较大,故

MFPL 策略的可管理性要低于 FMA 策略. 但从图 6 中可以看出, 两策略对可管理性的影响都比较稳定, 不会随着任务到达速率的变化而发生较大的变化.

6.2 随机替换策略和可预测替换策略

随机替换策略 RR 的特点是一个典型的“等待时间”的例子. 如果所有的节点有相等的平均服务周期, 由图 5 中可以看出, RR 策略的干扰不足其它固定或偏好优先策略干扰的两倍. 当 RR 在一次失效后, 选取一个节点 s_i , 失效的恢复时间不是能用一个简单的服务周期分布 l_2 来描述的. RR 将会随机选取一个当前可用的节点. 这将会使系统更偏好于那些有着更长服务周期的节点. 当某些节点提供过一次服务后, 其被 RR 策略选取的概率将被增加, 因为它有更多次机会被作为替换节点. 因此, 当任务到达速率较低时, 系统的平均失效率较低, 具有较长运行时间的节点立刻失效的可能性较小, 有较长运行时间的节点更容易被选取. 此时干扰对服务平均响应时间的影响程度较小, 故在任务到达速率较小时, 也就是当网格系统中, 处于服务状态的节点数目较小时, RR 策略可以为网格系统带来较高可管理性. 而随着任务到达速率的增加, 服务的平均失效率开始增加, 干扰对可管理性的影响也急剧增加.

本文选取研究的可预测替换策略是最小期望完成时间策略 SEAT, 与以上几种策略相比, 它可以通过当前系统的数据来预测未来系统的服务情况的变化情况. 因此我们可以从图 3、图 4 中看出, 此种策略较好地保持了系统的负载平衡, 且使系统服务具有较少的平均响应时间. 其在系统服务强度较小时, 可以保持较好的可管理性, 如图 6 所示, 但在干扰率较强的情况下, 其往往难以对系统未来情况进行准确的预测. 以至于可管理性急剧下降.

7 结 论

本文提出了网格服务的干扰和可管理性的定义及可量化的度量方法, 并建立了描述干扰发生过程的网格系统随机 Petri 网 (SPN) 模型, 为网格可管理性研究提供了参考模型. 同时本文对各种典型的服务管理策略进行了分类研究, 通过 SPN 模型的计算结果, 我们可以看出, 固定策略和偏好策略虽然在整体上表现出较差的性能, 但其具有较好的抗干扰能力, 随着任务强度的增加, 可管理性的波动较为平稳. 而随机替换策略和可预测替换策略在无干扰时, 具有较少的平均响应时间, 在有干扰发生但任务强

度较小时, 平均响应时间仍优于其它策略, 此时这两种策略具有较好的可管理性. 但当任务强度较大时, 干扰对其平均响应时间的影响较大, 导致服务的状态发生变化, 从而导致网格服务的可管理性严重下降. 本文的研究工作为具有高可管理性的网格服务管理策略的设计提供了依据. 下一步工作中我们将在服务管理策略中引入干扰因素, 从而实现在尽可能保证系统性能的情况下, 提高服务可管理性.

参 考 文 献

- [1] Foster I, Kesselman C, Tuecke S. The anatomy of the grid: Enabling scalable virtual organizations. *International Journal of Supercomputer Applications*, 2001, 15(3): 1-10
- [2] Foster I, Kesselman C, Nick J M, Tuecke S. The physiology of the grid: An open grid services architecture for distributed systems integration. <http://www.globus.org/research/papersogsa.pdf>, 2002
- [3] Foster I, Kesselman C, Lee C, Lindell B, Nahrstedt K, Roy A. A distributed resource management architecture that supports advance reservations and co-allocation//*Proceedings of the International Workshop on Quality of Service (IWQoS'99)*. London, UK, 1999: 27-36
- [4] Li J, Stribling J, Morris R, Kaashoek M F, Gil T M. A performance vs. cost framework for evaluating DHT design tradeoffs under churn//*Proceedings of the INFOCOM*. FL, USA, 2005: 225-236
- [5] Lin Chuang. *Stochastic Petri Nets and Performance Evaluation*. 2nd Edition. Beijing: Tsinghua University Press, 2005: 1-137 (in Chinese)
(林闯. 随机 Petri 网和系统性能评价. 第 2 版. 北京: 清华大学出版社, 2005: 1-137)
- [6] Shan Zhi-Guang, Lin Chuang, Ren Feng-Yuan. Modeling and performance analysis of a multiserver multiqueue system on the grid//*Proceedings of the 9th IEEE Workshop on Future Trends Distributed Computing Systems (FTDCS 2003)*. San Juan, Puerto Rico, 2003: 337-343
- [7] Maheswaran M. Quality of service driven resource management algorithms for network computing//*Proceedings of the International Conference on Parallel and Distributed Processing Technology*. Nevada, USA, 1999: 1090-1096
- [8] Hwang Soonwook. Grid workflow: A flexible framework for fault tolerance in the grid [Ph. D. dissertation]. California: Southern California University, 2003: 88-98
- [9] He X, Sun X, Gregor V L. QoS guided Min-Min heuristic for grid task scheduling. *Journal of Computer Science and Technology*, 2003, 18(4): 442-451
- [10] Weng C, Lu X. Heuristic scheduling for bag-of-tasks applications in combination with QoS in the computational grid. *Future Generation Computer Systems*, 2005, 21(2): 271-280

- [11] Schroeder Bianca, Gibson Garth A. A large-scale study of failures in high-performance computing systems//Proceedings of the International Conference on Dependable Systems and Networks (DSN2006). Philadelphia, PA, USA, 2006: 249-258
- [12] Meng Luo-Ming. Network management: Problems, progress and prospect. Journal of Beijing University of Posts and Telecommunications, 2003, 26(2): 1-8(in Chinese)
(孟洛明. 网络管理研究中的问题、现状和若干研究方向. 北京邮电大学学报, 2003, 26(2): 1-8)
- [13] De Roure D, Baker M A, Jennings N R, Shadbolt N R. The evolution of the grid. Grid Computing: Making the Global Infrastructure a Reality. John Wiley and Sons Ltd., 2003: 65-100
- [14] Bent J, Venkataramani V, LeRoy N, Roy A, Stanley J, Arpaci-Dusseau A, Arpaci-Dusseau R, Livny M. Flexibility, manageability, and performance in a grid storage appliance//Proceedings of the 11th IEEE Symposium on High Performance Distributed Computing. Edinburgh, Scotland, 2002: 3-12
- [15] Abdul Jabbar Mohammad, David Hutchison, James P G Sterbenz. Poster: Towards quantifying metrics for resilient and survivable networks//Proceedings of the 14th IEEE International Conference on Network Protocols (ICNP 2006). Santa Barbara, California, USA, 2006: 17-18
- [16] Godfrey P B, Shenker S, Stoica I. Minimizing churn in distributed systems. EECS Department, University of California, Berkeley: Technical Report EECS-2006-25, 2006
- [17] Laprie J-C, Kaaniche M, Kanoun K. Modeling computer systems evolutions: Non-stationary processes and stochastic Petri nets — Application to Dependability Growth//Proceedings of the 6th International Workshop on Petri Nets and Performance Models. Durham, NC, USA, 1995: 221-230
- [18] Geunmo K, Hyunsoo Y. On submesh allocation for mesh multicomputers: A best fit allocation and a virtual submesh allocation for faulty meshes. IEEE Transactions on Parallel and Distributed Systems, 1998, 9(2): 175-185
- [19] Sripanidkulchai K, Ganjam A, Maggs B, Zhang H. The feasibility of supporting large-scale live streaming applications with dynamic application end-points//Proceedings of the ACM SIGCOMM. Portland USA, 2004: 107-120
- [20] Wang Yuan-Zhuo, Lin Chuang, Zhai Zheng-Li, Yang Yang. A dependable task scheduling strategy for a fault tolerant grid model//Proceedings of the 1st International Workshop on Metropolis/Enterprise Grid and Applications (MEGA 2006). Lecture Notes in Computer Science 3842. Harbin, China, 2006: 534-539
- [21] Lin Chuang. Performance Evaluation of Computer Networks and Computer Systems. Beijing: Tsinghua University Press, 2001: 223-263(in Chinese)
(林闯. 计算机网络和计算机系统的性能评价. 北京: 清华大学出版社, 2001: 223-263)
- [22] Ciaodo G, Muppala J, Trivedi K S. SPNP: Stochastic Petri net package//Proceedings of the Petri Nets and Performance Models. Kyoto, Japan, 1989: 142-151



WANG Yuan-Zhuo, born in 1978, Ph. D., assistant researcher. His research interests include trusted network, grid computing, network QoS, and security evaluation.

LIN Chuang, born in 1948, Ph. D., professor, Ph. D. supervisor. His current research interests include computer

networks, performance evaluation, logic reasoning, and Petri net theory together with its applications.

YANG Yang, born in 1955, Ph. D., professor, Ph. D. supervisor. His current research interests include image processing and pattern recognition, TTS, STT, multimedia technology, network communication and Grid computing.

SHAN Zhi-Guang, born in 1974, Ph. D., professor. His current major research interests focus on computer network, Petri net work and system performance evaluation.

Background

This work is supported in part by a grant from the National Grand Fundamental Research 973 Program of China (No. 12003CB314804), and the National Natural Science Foundation of China (Nos. 90718040, 60673187, 60673054, 60673160, 60803123).

Grid is attractive and promising platforms for solving large-scale computing intensive problems, which realize a coordinated resource sharing and a problem solving in dynamic, multinstitutional virtual organizations. Various geographically distributed resources are logically coupled together and

presented as an integrated one. The heterogeneity and dynamism are traits of Grid. So the resource organizing and management become very complicated in a Grid environment. These make many churns to Grid service. And the quality of service (QoS) can not be always realized. A pervasive requirement of Grid systems is to improve the manageability and deal with churn. A high churn rate can increase costs or decrease service quality. And the formalized description and quantifying methods of Grid service churn and manageability are the preconditions.