

动态网格服务发现模型和算法研究

张 薇^{1,2} 李庆华¹ 马 丹³

(华中科技大学计算机学院 武汉 430074)¹ (武汉军械士官学校 武汉 430075)²

(武汉船用电力推进装置研究所 武汉 430064)³

摘 要 网格环境中,由于网格服务频繁加入、修改、撤销,使得动态网格服务发现成为一个困难问题。给出一个适用于服务动态发现的二维模型以及基于其上的服务注册、刷新和删除协议,并据此提出动态服务发现算法。算法采取局部集中式和全局分布式服务组织管理与发现机制,将服务功能与服务属性分开匹配,以适应频繁加入、撤出及注册信息刷新的动态服务发现。仿真结果从底层地理区域数目、网格规模、网络繁忙程度、网络动态程度等方面验证了模型与算法的优点及有效性。

关键词 网格,动态服务发现,模型,协议

中图法分类号 TP393 **文献标识码** A

Research on a Dynamic Grid Service Discovery Model and Algorithm

ZHANG Wei^{1,2} LI Qin-hua¹ MA Dan³

(College of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China)¹

(Wuhan Mechanical Technology College, Wuhan 430075, China)²

(Wuhan Institute of Marine Electric Propulsion, Wuhan 430064, China)³

Abstract Services join in and withdrawal from the grid frequently, this makes dynamic grid service discovery a difficult issue. This paper presented a two-dimensional model for dynamic service discovery. Based this model, the service's registration, refreshing, cancellation protocols and discovery algorithm were described. A partial centralized and global distributed service management and discovery mechanism was applied in the model, parting function and property matching on different nodes, boosting the holistic performance of dynamic grid service discovery. The simulation results show the model is working well in performance of dynamic grid service discovery according to such indexes as the scale of grid, the busy degree of grid, dynamic degree of grid, the partition granularity of grid.

Keywords Grid, Dynamic service discovery, Model, Protocol

1 引言

面向服务的网格体系结构将所有资源虚拟化为服务。现有网格服务或资源发现模型一般分为集中式和分布式两类^[1-6]。集中式服务发现模型存在扩展性问题,大量连接服务请求及动态的服务加入或撤出成为整个服务体系的瓶颈。分布式模型虽能解决扩展性问题以及动态服务频繁加入与撤出问题,但无法解决服务查找盲目性,即如何以正确方向查找所需服务。本文提出一个二维动态服务发现模型 TDMSD (Two-Dimensional Model of Service Discovery),给出 TDMSD 定义及基于模型服务注册、刷新和删除的协议与发现算法。模型根据服务功能和提供服务结点地理区域对网格服务进行划分,采用将服务功能与属性分开匹配的机制以适应频繁加入撤出及注册信息刷新的动态服务发现。采用服务属性性能排序机制,避免服务查找盲目性,保证了被发现服务在满足用户要求的情况下按某属性最优。

本文第2节给出 TDMSD 定义;第3节描述基于 TDMSD 服务注册、刷新与取消协议;第4节给出基于 TDMSD 服务发现算法;算法正确性证明以及复杂度分析在第5节给出;第6节给出仿真结果与分析;最后是总结和展望。

2 TDMSD 定义

2.1 服务资源描述

在结点上注册的服务信息用于服务发现,服务注册信息不涉及服务具体实现,因此面向服务发现的服务资源定义如下。

定义 1 面向服务发现的服务资源 s 定义为三元组 $s::=\langle GSH, P(s), D(s) \rangle$ 。

GSH(Grid Service Handle)为网格服务句柄,是服务唯一标识符,服务发现即找到满足要求的可用服务 GSH。 $P(s)$ 为服务属性集合,包括服务网络地址、提供服务结点地理位置、服务功能及时间参数等。 $D(s)$ 为某一时刻服务属性值集合。

到稿日期:2008-12-13 返修日期:2009-02-16 本文受国家自然科学基金(60503048)资助。

张 薇(1972—),女,副教授,主要研究方向为网格计算等,E-mail:msj0810@Tom.com;李庆华(1944—),男,博士生导师,主要研究方向为网格计算、并行计算;马 丹(1968—),男,博士,主要研究方向为任务调度。

2.2 模型中消息的描述

假设 m 为消息有限集合: $m::=\langle TTL, \text{message type}, \text{message sender}, \text{message receiver}, \text{message body} \rangle$ 。其中, TTL (Time of Live) 为生命周期, 消息类型 (message types) 有服务注册, 信息刷新, 取消, 操作成功确认, 发现请求, 发现超时, 发现失败和成功。在服务发现成功的消息中, 包含满足请求的 GSH。消息体包含服务功能和服务属性的相关数据。

2.3 TDMSD 描述

TDMSD 类似人类社会服务行业管理模型。在最底层, 均有一个最小不可分单位 (例如可以是校园、某企业) 来管理本单位所管辖区域内的服务。一方面, 在最底层, 若干地理位置相邻的小区域逐渐组成大的区域, 并设置地理区域管理机构进行管理, 如城市的社区组成区、区组成市。区政府和市政府即为地理区域管理机构。另一方面, 在最底层, 从属于每个地理位置区域内的服务根据其性质组成行业联盟。同一类服务属于同一个行业联盟, 同一个区域内有各种行业联盟。各不同行业联盟也依据地理位置划分逐渐形成更大范围内行业联盟, 以实现同类服务管理。在不同层次行业联盟设置行业联盟管理机构来管理。任何服务请求、注册、刷新、取消的消息都从最小不可分单位开始处理, 依次向上层结点转交处理。

如图 1 所示, TDMSD 中服务注册结点有 3 种: 基层管理结点 (Local Field Control Node, LFCN)、服务功能管理结点 (Service Function Control Node, SFCN) 和服务属性管理结点 (Service Property Control Node, SPCN)。模型首先将网格上所有注册的服务按提供服务结点所属最小地理区域进行划分, 设置 SFCN, 并在更大范围内设置更高层 SFCN, 直至一个总 SFCN 来管理全网格服务功能。服务功能依据公司与政府采购通用全球产品与服务的分类体系标准: 联合国标准产品及服务分类码 (United Nations Standard Products and Services Code, UNSPSC^[7])。而后, 模型对处于同一地理区域内服务按功能进行划分, 设置 SPCN, 并在更大范围内设置更高层 SPCN。功能相同的服务的属性均由同一个 SPCN 管理, 即一个区域内有多个 SPCN。在最底层, 由于所管理服务数目少, 将 SPCN 和 SFCN 合并为基层管理结点, 实现两种结点的功能, 在局部实现集中式管理。除最底层外, 所有服务功能都由 SFCN 管理, 属性由 SPCN 管理, 其功能与属性的管理与匹配分别由 SFCN 和 SPCN 完成, 在全局实现分布式管理。服务请求者可将其所处地理区域内任意一个 LFCN 作为服务发现入口, 提交服务请求。每个请求都有 TTL, 一旦 TTL 期满, 服务发现过程终止。

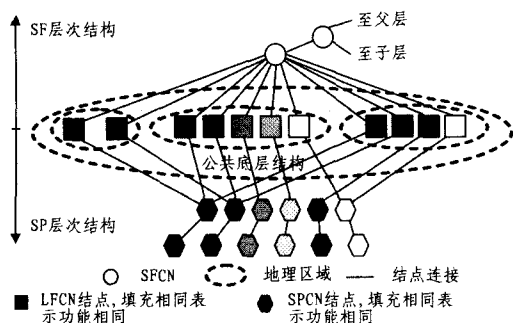


图 1 TDMSD

由 LFCN 组成的结构称为公共底层结构。LFCN 类似最

小不可分单位, 管理处于最小地理区域内所有服务的功能和属性。其管辖区域内所有服务均在其上进行注册发布, 并存储服务功能与属性信息。LFCN 完成的功能有对服务按照某项属性进行优劣排序、服务添加和删除、服务功能和属性的修改、接收服务请求者的服务请求、进行服务功能和属性匹配、决定服务请求的转发方向、等待父结点的服务发现结果, 并将结果返回服务请求者。LFCN 与其上的 SFCN 和 SPCN 两个父结点相连。

SFCN 组成结构称为 SF 层次结构, 用于构成树结构, 管理子 LFCN 和 SFCN 上的服务功能。其管理辖区内所有服务均在其上注册。SFCN 功能有: 完成服务添加与删除、服务功能修改、接收服务请求、进行服务功能匹配、快速判断服务请求的转发方向、接收其它结点服务发现结果, 并转发回请求结点。SFCN 与其父和子 SFCN 相连, 与其同层 SPCN 相连。

SPCN 组成结构称为 SP 层次结构, 用于组成树结构, 管理同层 SPCN 上具有相同功能的所有服务属性。其管理辖区内所有功能相同服务均在其上注册。SPCN 功能有: 完成服务添加与删除、服务属性添加删除和更新等修改、将所属相同功能服务按属性进行优劣排序、接收其同层 SFCN 转发服务属性匹配请求、将服务请求与所管服务进行属性匹配、判断服务是否可用、判断各属性是否满足要求、返回匹配结果。SPCN 与父和子 SPCN 相连, 与同层 SFCN 相连 (图中未画出)。

3 基于 TDMSD 的服务注册、刷新和取消协议

模型中, 服务任何改变都由 LFCN 接收处理并向上层结点转交。LFCN 接到服务改变消息后, 改变所管理的服务功能和属性, 将此消息向上层结点转交。上层结点接到服务改变消息后, 根据所管理类型进行相应操作。SFCN 管理服务功能, 改变相应服务功能。SPCN 管理服务属性, 则改变相应服务的属性。服务注册、刷新和取消协议描述如下。

(1) LFCN 收到服务提供者的服务注册消息, 将新服务根据某属性值插入到相应服务队列中。若接收到服务刷新消息, 则修改服务属性, 然后对服务队列重新排序。如果是取消消息, 则将服务从队列中删除。同时 LFCN 将功能信息向父 SFCN 转发, 属性信息向父 SPCN 转发。其协议过程如图 2(a) 所示。

(2) SFCN 接收到子 LFCN 或子 SFCN 发来的服务注册消息后, 一方面将此消息向父 SFCN 结点转发, 另一方面将此服务 GSH 和功能加入服务队列。如果是服务刷新消息, 则修改队列中相应服务的 GSH 和功能进行。如果收到服务取消消息, 则删除此服务。其协议过程如图 2(a), 图 2(b) 所示。

(3) SPCN 接收到子 LFCN 或子 SPCN 发来的服务注册消息后, 一方面将此消息向父 SPCN 结点转发, 另一方面将此服务按某属性值插入到已按此属性从优到劣排序的服务队列中。如果收到服务刷新消息, 在服务队列中修改相应服务属性, 并对服务队列重新排序。如果收到的是服务取消消息, 则删除此服务。其协议过程如图 2(a), 图 2(c) 所示。

接收服务注册、刷新与取消消息的结点操作成功后, 均向发消息的结点返回操作成功确认消息 (图中未画出)。如在规定时间内未收到操作成功确认消息, 则重复发送。

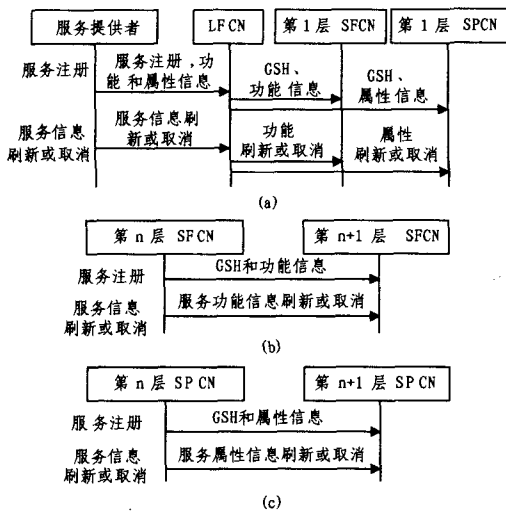


图2 结点间服务注册、刷新和取消协议

4 基于 TDMSD 服务发现算法

定义2 如果服务请求的功能和属性要求与服务的功能和属性均匹配成功,则称为服务发现成功。

下面将输入服务请求消息的结点记为 input。

4.1 结点的状态转换

如图3所示,结点有3种状态,并在相应触发事件下相互转换。其中,就绪为结点准备好接收消息的状态,消息可以是服务注册、刷新或取消以及服务请求消息。运行行为结点进行服务注册、信息修改、服务删除、排序、功能匹配或属性匹配时所处的状态。等待为结点转发消息后所处状态。

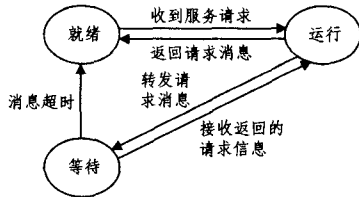


图3 结点状态转换

收到 input 的消息后,结点从就绪进入运行。将请求消息转发到其它结点后,结点从运行进入等待。接收到返回消息后,结点从等待回到运行。将消息返回给转发来消息的结点后,结点从运行进入就绪。转发或广播出的消息超时,结点从等待回到就绪。结点在就绪和等待时均可接收消息。结点在3种不同状态间转换,使整个模型以并行分布方式处理发现请求。

4.2 基于 TDMSD 的服务发现算法

基于 TDMSD 服务发现算法由4部分组成,描述如下。

(1)服务请求。服务请求者向属同一地理位置域内任意 LFCN 提交服务发现请求,等待接收返回消息。如返回消息是失败消息,则服务发现失败;如是 GSH,则服务发现成功。

(2)LFCN 结点搜索算法,简称 LFCN 算法,如图4所示。接收服务发现请求的 LFCN 对 input 发来的请求消息进行 TTL 判断后,将其要求与本结点所管理的服务进行功能匹配。如功能匹配成功,则在满足要求服务中从优到劣进行属性匹配。如属性匹配成功,则返回发现成功消息和 GSH 给请求者。如功能匹配不成功,或属性匹配不成功,则将请求消

息转发到父 SFCN,等待接收返回消息,将返回的消息转发给服务请求者。

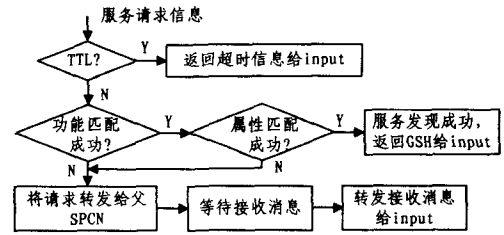


图4 LFCN 算法

(3) SFCN 结点搜索算法,简称 SFCN 算法,如图5所示。SF 层次结构中 SFCN 接收到子结点(LFCN 或下一层 SFCN)发来的请求消息进行 TTL 判断后,将请求要求与所管理的服务进行功能匹配。如果功能匹配成功,则将请求消息转发到与 SFCN 对应的本层 SPCN,等待接收返回消息。如本层 SPCN 返回发现成功消息,则返回成功消息及 GSH 给 input。如是失败消息,则判断有无父结点,如无父结点,则直接返回失败消息给 input。如有父结点,则将请求消息转发至父结点,并等待返回消息,将返回消息返还给 input。

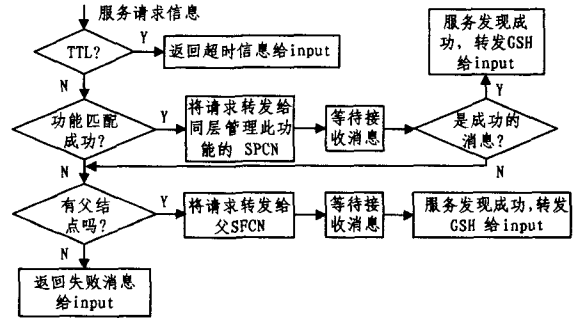


图5 SFCN 算法

(4) SPCN 结点搜索算法,简称 SPCN 算法,如图6所示。SP 层次结构中 SPCN 接收到 SFCN 发来的请求消息进行 TTL 判断后,在满足要求的服务中从优到劣进行属性匹配。如属性匹配成功,返回服务发现成功消息和 GSH 给 input。如属性匹配不成功,返回失败消息给 input。

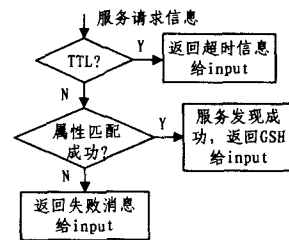


图6 SPCN 算法

算法中,所有结点接收到请求消息后均进行 TTL 判断,如 TTL 未超时,上述过程正常,如 TTL 超时,则返回失败消息给 input。各结点在等待接收过程中即可转入对其它消息进行处理。

5 模型和服务发现算法的正确性证明及复杂度分析

引理1 TDMSD 中,如在时间 t_0 到 t_1 之间,网络环境中网络状态有变化,在 t_1 后无变化,那么在一个有限时间后,网

格服务注册信息最新,据某项指标的网格服务排序也最新。

证明:TDMSD中,如服务提供者向它所属LFCN注册新服务或刷新已注册信息,在向LFCN发送消息时,如此时正处于 t_0 到 t_1 之间,则其信息无法到达,也收不到LFCN发回的确认消息,即注册或刷新失败。但 t_1 后,由于网络状态无变化,服务提供者所发送消息均能收到LFCN发回的确认消息,则注册或刷新成功,LFCN上服务信息是最新的。LFCN周期向其上层SFCN和SPCN进行注册信息的刷新。如正处于 t_0 到 t_1 之间,则消息无法正确传递,但 t_1 以后,LFCN服务刷新信息能正确传递到父SFCN。SFCN和SPCN也分别定期向父SFCN和SPCN发送刷新信息,此过程直至将刷新信息传递到顶层SFCN和SPCN。每层SFCN和SPCN在接收到刷新信息后,根据属性指标将此服务插入相应序列中。因此在有限时间后,网格服务注册信息最新,据某项指标的排序也最新。

定理1 TDMSD中,服务请求者在它所在地理区域内登录任意LFCN进行服务发现,如网格范围内无满足要求的服务,则一定查不到。如有满足要求的服务,则一定能查找到,且该服务按请求所需指标最优,距请求者地理位置最近,服务状态最新。

证明:网格范围内没有满足要求的服务分两种情况。

第一种情况:网格范围内不存在满足要求功能的服务。服务请求者向本区域LFCN发送请求消息。LFCN将消息中的功能属性与所管辖的服务进行功能匹配,有此功能的服务,则将请求转发到父SFCN,并等待接收返回消息。第一层SFCN接收到子LFCN转发的请求,与管辖服务进行功能匹配,无此功能的服务,将请求转发给父SFCN,直至总SFCN。由于网格范围内无满足要求的服务,失败消息沿原路返回到请求者。由于总SFCN管理全网格服务功能信息,请求者接收到失败消息说明网格范围内不存在满足要求功能的服务,请求者服务发现失败。

第二种情况:网格范围内有满足功能的服务,但属性不满足。请求者向它所在域内LFCN发送请求消息,过程同第一种情况,如任意SFCN上服务功能匹配成功,则转至对应同层SPCN上,进入SP层次结构进行属性匹配。SPCN将请求中属性与本结点所管理服务从优到劣进行属性匹配。如属性匹配失败,则服务匹配不成功,返回失败消息给input。该过程在SP和SF层次结构中重复执行,直至将失败消息发送回请求者,说明网格范围内无满足要求服务,服务发现失败。

网格范围内有满足要求的服务:请求者登录同域内LFCN,在SF层次结构中服务查找过程同第一种情况,如在任意结点上功能匹配成功,则转至相应结点同层SPCN上。SPCN从优到劣进行服务属性匹配,如服务属性匹配成功,则服务匹配成功,且此服务是本域内按照某项指标最优。根据引理1,SPCN所维护的网格信息最新,因此服务匹配成功的服务其状态最新。又因发现请求是从最底层逐渐向上层转发,即服务发现范围是以本地理区域为中心逐渐向外的,因此匹配成功的服务是距请求者地理位置最近的服务,也是按某项属性最优,服务状态最新。

引理2 SPCN算法时间复杂度是 $O(\log n)$,且是最优的。

证明:SPCN算法在已排好序的服务中顺序进行服务匹

配,是以比较为基础的有序检索问题,模型采用二分检索,算法时间复杂度为 $\log n$ 。设 $A(1:n)$ 含有 $n(n \geq 1)$ 个不同元素,排序为 $A(1) < A(2) < \dots < A(n)$ 。又设用以比较为基础的算法去判断是否 $x \in A(1..n)$,则算法在最坏情况下所需最小比较次数 $\text{FIND}(n)$ ^[8],从模拟求解检索问题算法的较树可知, $\text{FIND}(n)$ 不大于树中由根到一个叶子的最长路径的距离。而所有树中必定有 n 个内结点与 x 在 A 的 n 中可能的出现相对应。如果一棵二元树所有内结点所在级数小于或等于 k ,则最多有 $2k-1$ 个内结点。故 $n \leq 2k-1$,即 $\text{FIND}(n) \geq \log(n+1)$ 。故SPCN算法是TDMSD中最优算法。

引理3 SFCN算法时间复杂度是 $O(\log n)$ 。

证明:SFCN算法中,如果服务匹配成功返回GSH给input,时间复杂度为 $O(k)$ (k 为常数);如果服务匹配不成功,且不是顶层结点,则将请求发给父SFCN,时间复杂度为 $O(\log n)$ 。

引理4 LFCN算法时间复杂度是 $O(n * \log n)$ 。

证明:LFCN算法查找服务功能,算法最好的情况是一次找到,则时间复杂度为 $O(1)$;一般情况如input未超过TTL则:如功能匹配成功,则进行属性匹配,属性匹配成功时返回GSH给input,时间复杂度为 $\log n$;不成功则将请求发往父SFCN,调用SFCN算法,时间复杂度为 $O(n * \log n)$;如功能匹配失败,将请求发往父SFCN,调用SFCN算法,时间复杂度为 $O(n * \log n)$;

综上,LFCN算法时间复杂度是 $O(n \log n)$ 。

定理2 基于TDMSD服务发现算法时间复杂度为 $O(n \log n)$ 。

证明:由引理2、引理3、引理4可得,基于TDMSD服务发现算法时间复杂度为 $O(\log n) + O(\log n) + O(n * \log n)$,即为 $O(n * \log n)$ 。

定理3 基于TDMSD服务发现算法空间复杂度为 $O(n^2)$ 。

证明:证明方法和分析过程与定理2类似。

由定理2和定理3可知,基于模型的服务发现算法时间复杂度上有优势,空间复杂度较大。但对网格环境而言,由于其存储是分布的,因此算法空间开销是可以容忍的。

6 仿真结果与分析

本节给出基于NS-2仿真结果。假设父子结点间延迟为20ms,兄弟结点间延迟为10ms,结点间带宽为2M,lambda为用户发送请求信息动态程度,lambda值越小,用户发送请求信息越频繁。 T 表示请求信息得到成功回复时间。User表示服务请求者,Server表示结点。仿真结果如图7至图10所示。

从图7看出,在底层地理区域数目较少时, T 延迟较大,这是因为每个公共底层结点负责服务较多,需处理时间较多。当底层地理区域数目超过1000后, T 趋于平缓,这是因为底层地理区域User和Server分布均匀,在Server能够满足User情况下,公共底层结点没有过多负载。因此TDMSD中地理区域数目对服务发现的性能影响较小,具有较好的网络可扩展性能。从图8和图9看出,网络规模从500个结点增加到5000个时, T 并未随之剧烈变化。图9中绝大部分请求信息是通过公共底层的父结点得到成功回复,但在5000个结点

时 T 较 500 个结点略显增加,这是因为规模较大网格中的功能结点要处理更多消息。总体上,TDMSD 能适应不同网格规模情况,不会因网格规模增大而变得效率低下。图 10 说明当 λ 为 3 时,请求比较频繁,当 λ 为 11 时,请求较稀疏, λ 值由 3 变为 11 时网格动态程度发生较大变化,但 T 并未剧烈变化。

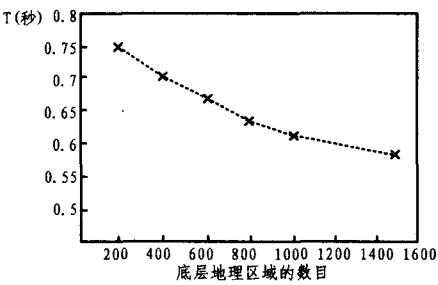


图 7 底层地理区域数目对 T 的影响

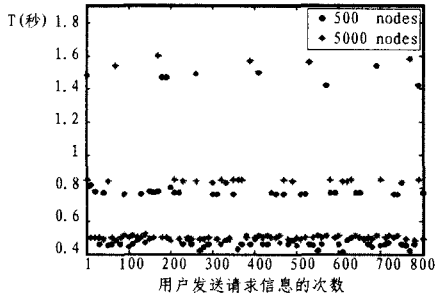


图 8 网格规模对 T 的影响

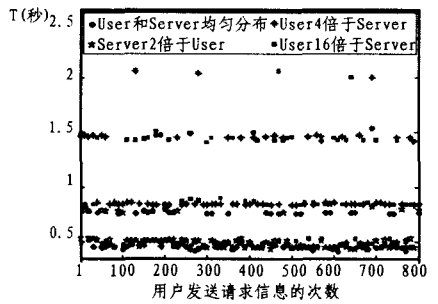


图 9 网络繁忙程度对 T 的影响

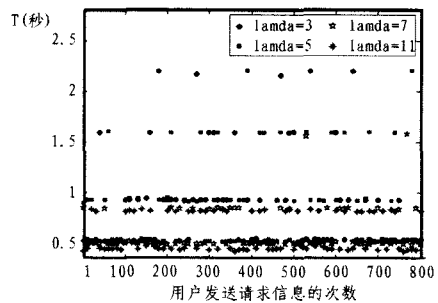


图 10 5000 个结点情况下网格动态程度对 T 的影响

通过上面仿真数据得到表 1 所列的结果。当 User 分布不均匀时 T 差别较大,但当网格中 User 达到一定数目后, T

均值变化将会很缓慢。在 User 提交请求信息的动态程度不同时 T 的均值在 λ 为 3 和 5 时要大于为 7 和 9 时的情况,但差距并不大。User 提供请求信息的频繁程度在 λ 小于 5 后 T 均值变化很小。

表 1 请求信息得到成功回复的时间均值

5000 个结点	User /Server				λ			
	1/2	1	4	16	3	5	7	11
T 均值(秒)	0.53	0.66	0.92	0.97	0.87	0.85	0.61	0.598

结束语 本文给出一个二维服务发现模型以及基于该模型服务注册、刷新、取消协议和动态服务发现算法。证明了其正确性,给出了复杂度分析和仿真结果。模型兼具集中式服务发现模型的全局信息获取上的优势以及分布式服务发现模型复杂度低、对服务动态变化适应性的优点,同时克服了集中式模型所固有的不可扩展性、复杂度高开销大的问题以及分布式模型的服务发现盲目性的缺点。由于服务加入、删除和刷新信息只向结点的上一层结点传递,因此这些服务更新信息的改变只会改变局部结点所维护的信息的更改,而不会波及到全网范围。即使服务频繁地动态变化也不会引起信息风暴。

进一步的工作是在已有模型基础上引入如基于经验、基于用户爱好等的服务发现机制以进一步提高服务发现的性能。

参考文献

- [1] Messig M, Goscinski A. Autonomic System Management in Mobile Grid Environments[C]// Proceedings of The 5th Australasian Symposium on Grid Computing and e-Research, Australia, 2007
- [2] Heine F, Hovestadt M, Kao O. Towards ontology-driven P2P grid resource discovery[C]// Proceedings of Fifth IEEE/ACM International Workshop on Grid Computing, Los Alamitos, CA, USA, 2004:76-83
- [3] 武秀川,鞠九滨. 计算网格的资源分发和发现机制[J]. 计算机科学, 2003, 30(1):12-13
- [4] Derbal Y, et al. Confidence-based grid service discovery[J]. International Journal of Web and Grid Services, 2008, 4(2):189-210
- [5] LI Maozhen, et al. Grid Service Discovery With Rough Sets[J]. IEEE transactions on knowledge and data engineering, 2008, 20(6):851-862
- [6] Weil S A, Brandt S A, et al. Grid resource management-CRUSH: controlled, scalable, decentralized placement of replicated data [C] // Proceedings of the 2006 ACM/IEEE conference on High Performance Networking and Computing, Florida, 2006
- [7] <http://www.unspsc.com>
- [8] Horowitz E, Sahni S. Fundamentals of Computer Algorithms [M]. Pitman, Inc: Computer Science Press, 1978:78-80