

资源定位服务的分布式生成树模型及算法研究

王铁军, 刘 恒, 孙 明, 周明天

(电子科技大学计算机科学与工程学院, 四川成都 610054)

摘 要: 针对云计算中资源定位索引的瓶颈问题, 基于分布式生成树 DST 提出了一种自适应的资源定位服务 DST-RLS 模型以及搜索半径限制的资源定位算法 SRL. 通过给出代表元选取规则, 使代表元均衡分布在 DST-RLS 模型中, 提高了资源定位服务的负载均衡性. SRL 算法支持复杂查询和灵活的搜索策略, 算法的时间复杂度为常数阶. 数学分析和仿真实验表明, DST-RLS 模型具有较好的负载均衡性、可靠性和自适应性; 通过限制搜索半径, SRL 算法缩短了资源定位时间, 减少了网络带宽耗用, 性能优于 DST 定位算法.

关键词: 资源定位; 分布式生成树; 对等网络; 云计算; 数据网格

中图分类号: TP393 **文献标识码:** A **文章编号:** 0372-2112 (2011) 02-0364-06

Research on the Model and Algorithms Based on Distributed Spanning Tree for Resource Location Service

WANG Tie-jun, LIU Heng, SUN Ming, ZHOU Ming-tian

(Department of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, Sichuan 610054, China)

Abstract: A self-adaptive resource location service model (DST-RLS) based on distributed spanning tree (DST) and search radius limited resource location algorithms SRL were proposed to resolve the bottleneck of the resource location index in the cloud computing. The rule of selecting representative was also presented to make the representatives distributed uniformly in the DST-RLS model, and which improves the load balance of the resource location service. The SRL algorithm supports the complex queries and has time complexity of constant level. The mathematical analysis and simulation results show that the DST-RLS model provides good load balance, reliability and self-adaptive; by limiting the search radius, the SRL algorithm can reduce the search time, lower the network bandwidth usage, and whose performance is better than DST resource location algorithm.

Key words: resource location; distributed spanning tree (DST); P2P networking; cloud computing; data grid

1 引言

随着应用需求的发展和网络计算机技术的进步, 云计算作为一种新的主流计算模型得到越来越多科研机构和企业的支持. 众多与云计算相关的产品、公共协议和开放平台应运而生, 如 IBM 的蓝云^[1]、清华大学的透明计算^[2]和芝加哥大学的 Nimbus 项目^[3]. 作为云计算基础平台的云存储服务, 为云计算提供大规模数据存储和处理能力. 云存储平台通常采用分块和副本的方式对资源进行管理, 并将存储物理资源映射信息的逻辑资源索引存入资源副本编目. 云存储通过资源定位服务查找所需的逻辑资源索引, 并转换为物理资源提供给

上层应用访问.

在分布式计算中, 资源副本编目有三种组织形式: (1)中心副本编目. 谷歌 GFS^[4]通过单一的主服务器存储和管理资源副本信息, 并采用平衡二叉树方式提供元数据的索引服务. 单一主服务器是系统性能的瓶颈, 降低了系统的可扩展性和可靠性; (2)分级副本编目. 在数据网格中, 欧洲数据网格项目^[5]采用多根树的形式建立分级副本编目提供资源定位服务. 但在处理大规模请求时同样存在系统瓶颈, 此外由于采用人工静态配置分级副本编目的形式降低了系统的可扩展性和易用性; (3)基于 P2P 覆盖网络的副本编目. 文献^[6]和^[7]中给出了基于分布式哈希表 (DHT) 的自适应资源定位服务, 副本

编目结点自组织成 DHT 网络,增强了系统的自适应性和可靠性.其中,DHT 网络中资源定位的时间是 $O(\log N)$.但由于哈希操作破坏了资源副本元数据信息的可读性,故无法支持复杂搜索.

针对云计算中资源定位索引的瓶颈问题,本文基于分布式生成树^[8](Distributed Spanning Tree, DST)提出一种自适应的资源定位服务模型 DST-RLS 及限制搜索半径的资源定位算法 SRL. DST-RLS 模型支持复杂搜索,具有较好的容错性和负载均衡性. SRL 算法支持四种搜索策略可提供灵活的资源定位服务.通过数学分析和实验仿真对 DST-RLS 模型和 SRL 算法的性能进行了评估,最后给出本文结论.

2 DST-RLS 模型

DST-RLS 模型是一种基于 DST 的非结构化纯对等网络.在创建资源副本时,从物理资源到逻辑资源的映射信息被存入 DST-RLS 结点的本地资源编目(LRC)中. DST-RLS 结点通过结点组织管理算法自组织成覆盖网络,向上层应用提供资源定位服务,如图 1 所示.逻辑资源索引包含资源副本的元数据信息和访问控制信息.为维护 DST-RLS 模型和资源状态信息的一致性, DST-RLS 结点需运行副本状态更新服务,以推的方式周期性地向邻居结点发送结点状态信息和逻辑资源索引更新信息.

2.1 DST-RLS 模型的形式化表述

DST-RLS 模型基于 DST 结构.逻辑上, N 个结点构成的 DST-RLS 模型可分成 $H = O(\log N)$ 层, H 是 DST-RLS 模型的高度. N 个结点在第 i 层($1 \leq i \leq H$)被分成 m_i 个组,记作: $g_{i,1}, g_{i,2}, \dots, g_{i,m_i}$. 第 1 层组的元素是结点本身;当 $i \geq 2$ 时,第 i 层组 $g_{i,j}$ 的元素是第 $i-1$ 层组. 组 $g_{i,j}$ 的元素个数为 $e_{i,j}$,记作 $|g_{i,j}|$;组 $g_{i,j}$ 中结点个数为 $n_{i,j}$,记作 $\|g_{i,j}\|$. 第 i 层的 m_i 个组称为第 i 层组,并将第 1 层和第 H 层组分别称做底层组和顶层组.根据 DST 的定义有: $0 \leq e_{H,j} \leq b$ 和 $a \leq e_{i,j} \leq b$,且 $0 \leq n_{H,j} \leq N$

和 $a^i \leq n_{i,j} \leq b^i$.此时,称 a 和 b 分别是 DST-RLS 模型的稀疏值和饱和值.用符号 $n_k(g_{i,j})$ 表示结点 n_k 在组 $g_{i,j}$ 中.为缩小结点路由表,路由表只记录其它组中的一个结点的信息,称这个结点为代表元结点.符号 $n_j = n_i \uparrow g_{k,m}$ 表示结点 n_i 选择结点 n_j 作为第 k 层组 $g_{k,m}$ 的代表元.结点 n_k 路由表 t_k 分层存放后继结点 $t_k \cdot S$ 和前驱结点 $t_k \cdot P$,其中 $t_k \cdot S = \{s | s = n_k \uparrow g_{l,m}, 1 \leq l \leq H\}$, $t_k \cdot P = \{p | n_k = p \uparrow g_{l,n}, 1 \leq l \leq H\}$.为实现结点间的通信,模型中所有组内元素互连形成团(Clique).根据结点路由表信息,每个结点 n_k 均可构造一棵以自己为根的倒叉树 T_k .由 N 个结点组成的 DST-RLS 模型,逻辑上存在 N 棵高度是 H 的倒叉树.图 1 中由实线连接的组构成结点路由表;虚线箭头指向第 i 层代表元结点;由实线和虚线所连接的组共同构成倒叉树.

若 $\Sigma = \{\omega | \omega \in \mathbf{N} \wedge 1 \leq \omega \leq b\}$ 表示有限长度的字母表,则符号 Σ^* 表示所有构建在 Σ 上有限长度的字符串集合.若用下标(从 1 开始)标记每一个组成字符串 s 的字符 s_i ,则字符串 s 可写成 $s_1 s_2 s_3 \dots$,即 $s[i] = s_i$.用符号 $|s|$ 表示字符串 s 的长度.若 $|s| = 0$,则字符串 s 为一个空串,用符号 φ 表示.若 $|s| \geq i$,则符号 $s \downarrow i$ 表示字符串 s 长度为 i 的后缀.若 $s \downarrow i = w$,则表示字符串 w 为字符串 s 长度为 i 的后缀,记为 $s \geq w$.若字符串 s 和 v 有共同的最长后缀 w ,记作: $s \nabla v = w$. DST-RLS 模型中结点路由表的代表元结点有序存放,并通过索引值表示代表元所在组的位置.若 $x \in g_{k,m}$,则符号 $g_{k,m}[l] = x$ 表示结点 x 在组 $g_{k,m}$ 中的位置是 l .当结点 x 加入模型时,系统会自动为其生成全局唯一的 ID 标识符,并根据所在位置分配一个关系向量 $v(x)$.关系向量 $v(x)$ 由下向上依次记录结点 x 在其路由表每层组中的位置.关系向量会随着覆盖网络结构的变化而改变,但在某一时刻全局唯一.根据上述字母表和字符串的定义, DST-RLS 模型中任意结点 x 和 y 的关系向量满足: $v(x), v(y) \in \Sigma^*$,且 $|v(x)| = |v(y)| = H$.若同时满足如下两个条件:(1) $\exists k \in \mathbf{Z}$,使 $|v(x) \nabla v(y)| = H - k$; (2)若 $k \geq 2, \forall 1 \leq l < k$,若 $y \in g_{l,i}$,有 $(v(x)[l] + v(x)[l+1]) \equiv v(y)[l] \pmod{|g_{l,i}|}$,则称结点 x 在 k 级相似于结点 y ,记作 $x \xrightarrow{L=k} y$.其中, $v(x)[l+1]$ 是协调因子,起到均衡代表元的作用.

2.2 模型的负载均衡性

若结点 n_i 第 l 层路由表满足: $|t_k \cdot P[l]| \gg |t_k \cdot S[l]|$,则称结点 n_i 为关键结点.由于存在相对较多的前驱结点,关键结点的负载通常大于普通结点.同时,关键结点的存在降低了模型的容错性.

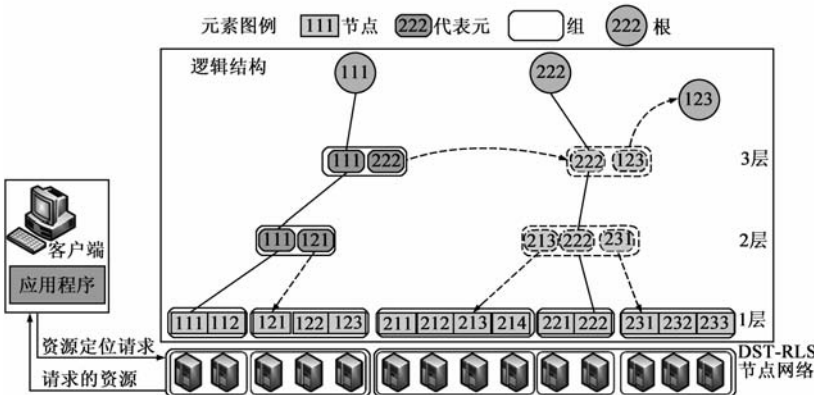


图1 DST-RLS结点组成的覆盖网络

DST-RLS 模型引入了代表元选取规则:对于 DST-RLS 中任意两个结点 $x \in g_{k,m}$ 和 $y \in g_{k,n}$, 若 $y = x \uparrow g_{k,n}$, 当且仅当 $x \xrightarrow{L=k} y$ 成立. 代表元选取规则使得代表元结点均匀分布在模型中, 从而降低了关键结点出现的概率, 增强了系统的负载均衡性和容错性.

定理 1 在 DST-RLS 模型中, 任意第 k 层组中结点被所有同层组中结点选作代表元的概率相差不超过 $1/a(a-1)$.

证明 首先考虑一个第 k 层组 $g_{k,i}$ 的情况: 对任意结点 $x \in g_{k,i}$ 和 $y \in g_{k,j}$, 有 $y = x \uparrow g_w$. 不失一般性, 设 $v(x)[k+1] = o, |g_{k,i}| = m, |g_{k,j}| = n$, 则有 $a \leq o, m, n \leq b$. 若 $m \bmod n = r$ 且 $\lfloor m/n \rfloor = q$, 则结点 y 被结点 x 选作代表元的概率 P 与结点 y 在组 $g_{k,j}$ 中的位置 $l = v(y)[k]$ 有关. 若 $L = (l-1+o) \bmod n$, 则概率 P 满足:

$$P = \begin{cases} \frac{q}{m}, & L=0, L > r \\ \frac{q+1}{m}, & 0 < L \leq r \end{cases} \quad (1)$$

则组 $g_{k,j}$ 中不同结点被同层其它组 $g_{k,i}$ 中结点选作代表元的概率相差最大值满足:

$$P'_{g_{k,i}} = P_{0 < L \leq r} - P_{L=0, L > r} = \frac{q+1}{m} - \frac{q}{m} = \frac{1}{m} \quad (2)$$

下面考虑所有第 k 层组的情况. 不失一般性, 假设第 $k+1$ 层组 G_{k+1} 由 s 个第 k 层组 $\langle g_{k,1}, g_{k,2}, \dots, g_{k,s} \rangle$ 组成, 其中 $a \leq s \leq b$. 计算第 k 层组 $g_{k,1}$ 中结点被所有同层其它组中结点选作代表元的概率差. 由于引入了协调因子, 设所有第 k 层其它组 $g_{k,2}, \dots, g_{k,s}$ 中结点组成新组 G'_{k+1} , 则有:

$$|G'_{k+1}| = |g_{k,2}| + |g_{k,3}| + \dots + |g_{k,s}| = \sum_{i=2}^s |g_i| \quad (3)$$

不失一般性, 设结点路由表中每组代表元的平均数为 σ , 其中 $a \leq \sigma \leq b$. 故 $|G'_{k+1}| = \sigma(\sigma-1) \geq a(a-1)$. 将式(3)代入式(2)有:

$$P'_{G'_{k+1}} = \frac{1}{\sum_{i=2}^s |g_i|} = \frac{1}{\sigma(\sigma-1)} \leq \frac{1}{a(a-1)} \quad (4)$$

故被所有同层组中结点选作代表元的概率相差不超过 $1/a(a-1)$. \square

定理 1 说明, 代表元选取规则可使代表元尽可能均衡地分布在模型中, 降低了关键结点出现的概率. 此外, 由于每个结点均可处理资源定位请求, 从而将本来由一个或几个资源定位索引结点承担的负载均摊到每个结点上, 故 DST-RLS 模型的负载均衡性优于使用中心副本编目和分级副本编目的资源定位系统.

3 资源定位算法

文献[8]中给出了 DST 算法, 该算法使用 $O(\log N)$ 时间可定位到目标资源, 性能与 DHT 查找算法相当. 同时, 文献[9]的研究结果表明, 在同等条件下 DST 算法的平均搜索时间及其可扩展性优于图和二叉树的搜索算法. DST 算法通过限制 TTL 控制算法的搜索深度, 但无法细粒度控制搜索范围, 算法的冗余开销依然很大. 为此, 本文提出了一种搜索半径限制的资源定位算法 SRL, 可有效减少带宽耗用并缩短搜索时间.

3.1 基于搜索半径的定位算法

定义 1 搜索深度: 在 DST-RLS 模型中, 若结点 n_i 的一次资源定位从路由表的第 i 层 $t_{k,i}$ 开始到第 j 层 $t_{k,j}$ 结束, 则将 $d = |i-j|$ 定义为本次资源定位的搜索深度.

定义 2 搜索广度: 在 DST-RLS 模型中, 若资源定位过程发现一条从发起结点到目标结点的路径 p , 则将路径长度(路径上结点的数目) $b = |p|$ 定义为本次资源定位的搜索广度.

定义 3 搜索半径: 在 DST-RLS 模型中, 请求发起结点 n_i 将对搜索深度 $d \leq R$, 搜索广度 $b \leq R$ 范围内的结点进行遍历, 此时将 R 定义为 DST-RLS 模型的搜索半径.

根据搜索方向和并发性的不同, SRL 算法可支持四种搜索策略: 串行本地优先 (SEQ-LFS), 串行远端优先 (SEQ-RFS), 并行本地优先 (PAL-LFS) 和并行远端优先 (PAL-RFS). 在处理一条资源定位请求消息时, 串行方式在同一时刻只有一个结点在处理和转发该请求; 而并行方式则同时向路由表某一层中所有代表元转发该请求. LFS 从路由表的底层组开始向上层组广播, 优先遍历逻辑上距离较近的本地结点; 而 RFS 则从顶层组向下层组广播, 优先遍历逻辑上距离较远的远端结点. SRL 算法结束的条件是定位到有效结果数量的资源、搜索达到模型高度 H 或搜索半径 R . 在实际环境中, SRL 算法可根据不同的应用场景灵活地使用不同的搜索策略提供差异化的服务.

3.2 定位算法的性能分析

定理 2 DST-RLS 模型中, 相同搜索半径的 RFS 可遍历结点数大于 LFS, SRL 算法时间复杂度是 $O(1)$.

证明 从任意结点 n_i 开始的资源定位过程, 最坏情况下将遍历搜索半径 R 范围内的全部结点. 不失一般性, 设结点路由表中每层代表元个数平均为 σ , 其中 $a \leq \sigma \leq b$. LFS 算法将自下向上依次对以路由表第 k 层 σ 个代表元为根、高为 k 的子树上的所有结点进行搜索, 其中 k 取值从 1 到 R . 因为每棵高度为 k 的子树上结点个数为 $TN_k = \sigma^k$, 故 LFS 算法可遍历结点数 $TN_{LFS} =$

$\sum_{k=1}^R TN_k = \sum_{k=1}^R \sigma^k$, 满足 $\sum_{k=1}^R a^k \leq TN_{LFS} \leq \sum_{k=1}^R b^k$. RFS 算法将自上向下依次对以路由表第 k 层 σ 个代表元为根子树的上面 R 层中结点进行搜索, 其中 $H \geq k \geq H - R + 1$. 因为每棵子树上面 R 层中结点个数为 σ^R , 故 RFS 算法可遍历结点数 $TN_{RFS} = \sum_{k=H-R+1}^R \sigma^k = R \cdot \sigma^R$, 满足 $R \cdot a^R \leq TN_{RFS} \leq R \cdot b^R$.

因为 $b \geq 2a > 2$, $R \cdot b^R \geq \sum_{k=1}^R b^k$, 故搜索半径 R 的 RFS 算法可遍历结点数大于 LFS 算法. 此外, SRL 算法最多遍历结点数 $TN_{max} = \max\{TN_{LFS}, TN_{RFS}\} = R \cdot b^R$. 设搜索每个结点的平均时间为 τ , 那么定位算法最长搜索时间 $t_{max} = \tau \cdot TN_{max} \leq \tau \cdot R \cdot b^R$. 由于搜索时间中三个参数的取值与网络规模 N 无关, 故 SRL 算法的时间复杂度为 $O(1)$. \square

定理 2 的证明过程说明, 缩小搜索半径可减少搜索时间. 与对等网络不同, 为了提供可靠一致的云存储服务, 云计算须关注热点资源和非热点资源. 同时, Lv 等人实验对比了均匀分布、成比例分布和平方根分布环境下搜索算法耗用资源的情况. 实验结果显示, 资源均匀分布时带宽资源耗用最高、网络延迟最大^[10]. 因此, 本文讨论资源副本均匀分布的 DST-RLS 模型. 设逻辑资源索引信息以概率 p 均匀分布在模型中, SRL 算法在找到 1 个满足条件的逻辑资源索引后停止搜索.

定理 3 在逻辑资源索引以概率 p 均匀分布的 DST-RLS 模型中, 若需以概率 F 成功定位目标资源, 则 LFS 搜索半径满足 $R \geq \log_{\sigma}[(\sigma - 1) \cdot k / \sigma + 1]$; RFS 搜索半径满足 $R \geq W(k \cdot \ln \sigma) / \ln \sigma$. 其中 σ 为路由表中每组代表元的平均数 ($a \leq \sigma \leq b$), $k = \log_{1-p}(1 - F)$.

证明 逻辑资源索引以概率 p 均匀分布在模型中, 则在每个结点处找目标资源的概率是 p . 设停止搜索时遍历结点数是 X , 那么在第 k 个结点上定位到目标资源索引的概率是:

$$P\{X = k\} = p(1-p)^{k-1}, k = 1, 2, \dots \quad (5)$$

设无论是否定位到目标资源索引, 遍历 Y 个结点后停止搜索. 则由式(5)可知, 遍历 k 个结点后定位到目标资源索引的概率是:

$$\begin{aligned} P\{Y = k\} &= \sum_{i=1}^k P\{X = i\} \\ &= \sum_{i=1}^k p(1-p)^{i-1} = 1 - (1-p)^k \end{aligned} \quad (6)$$

由式(6)可知, 需以概率 F 定位到目标资源索引需遍历结点数 k 满足:

$$k = \log_{1-p}(1 - F) \quad (7)$$

不失一般性, 设 DST-RLS 模型结点路由表每组中代表元的平均数为 σ ($a \leq \sigma \leq b$).

由定理 2 可知, LFS 算法可遍历最大结点数 $TN_{LFS} = \sum_{k=1}^R \sigma^k$, RFS 算法可遍历最大结点数 $TN_{RFS} = R \cdot \sigma^R$. 若以概率 F 成功定位到目标资源, 对于 LFS 搜索半径 R 应满足 $TN_{LFS} = \sum_{k=1}^R \sigma^k \geq k$, 则有:

$$R \geq \log_{\sigma} \left[\frac{(\sigma - 1)k}{\sigma} + 1 \right] \quad (8)$$

对于 RFS 搜索半径 R 应该满足 $TN_{RFS} = R \cdot \sigma^R \geq k$, 则有:

$$R \geq \frac{W(k \ln \sigma)}{\ln \sigma} \quad (9)$$

其中函数 W 是 LambertW 函数. \square

4 仿真实验

仿真程序使用 Python 编写, 运行在 Intel T7500 2.2GHz CPU, 2G 内存, OS 是 OpenSuSE 10 的电脑上.

实验 1 模型的自适应性仿真实验, 考察自适应算法的时间复杂度.

本实验分两个部分, 分别模拟结点加入和离开覆盖网络的情况. 记录试验中产生的消息数, 对比消息平均数、模型高度和网络规模间的关系. 其中, 模型参数 $a = 2$, $b = 4$, 实验未考虑结点处理和网络负载能力.

由图 2 可知: (1) 结点加入和离开模型时发送消息的平均数随网络规模 N 的增加成对数级增长. 设每条消息的处理时间相同, 则实验结果表明结点管理算法的时间复杂度为 $O(\log N)$; (2) 在模型高度不变的情况下, 随着结点数目的增加, 结点路由表中每层代表元数目向饱和值 b 增长, 结点加入或离开时所需发送消息数也随之增加; (3) 对比图 2(a) 和图 2(b) 发送消息平均数的纵坐标发现, 在相同网络规模和模型高度的情况下, 结点加入模型需发送消息的平均数基本上是结点离开模型时的两倍. 原因是结点在离开覆盖网络时, 只需向路由表中代表元结点发送离开消息, 省去了通过介绍结点收集覆盖网络信息的过程. 以上两个过程所需发送消息的平均数基本相当, 故结点加入模型所需发送消息平均数基本上是结点离开模型时的两倍.

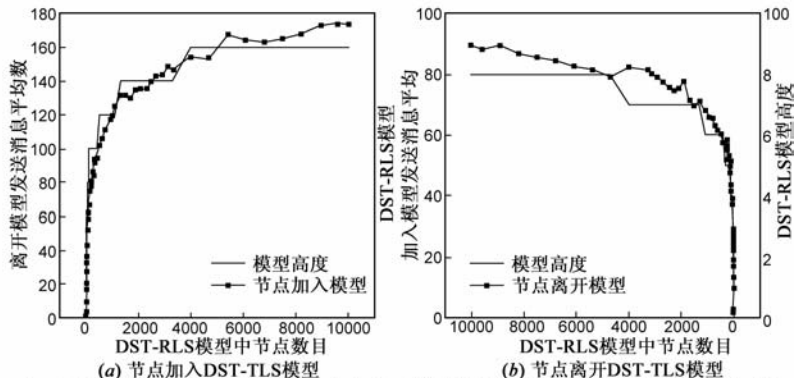


图 2 不同规模覆盖网络下结点加入和离开模型发送消息平均数和模型高度的对比

实验 2 对比四种资源定位算法的性能,给出适合使用的场景。

程序模拟客户端随机向模型中结点发送资源定位请求,对比 SRL 算法四种资源定位策略的平均搜索时间、网络带宽使用和搜索结果随网络规模的变化情况。其中,模型参数 $a=4, b=8$, 搜索半径 $R=3$, 逻辑资源索引以 0.05 的概率均匀分布在模型中, 每条消息的处理延迟为 200ms。在 1 次仿真实验中, 客户端平均每秒发起 2.4 个请求, 实验持续 5min。每种定位策略分别进行 10 次仿真实验, 结果如图 3 所示。

通过图 3 对比四种搜索策略的性能指标可知: SEQ-LFS 和 SEQ-RFS 整体性能相当, 对于系统和网络资源的使用最少, 可在模型高负载或为试用用户提供服务时使用。同时, 模型可根据 LFS 和 RFS 的区别, 实现对逻辑上不同区域内资源的定位和使用。此外, PAL-RFS 可在一个资源定位请求中定位更多结果, 但却消耗大量系统和网络资源, 可考虑在向注册或付费用户提供服务时使用。最后, 与其它三种搜索策略相比, PAL-LFS 是一种折中方案, 在搜索时间、带宽占用和搜索结果均有不错的表现, 其应用更具普遍性。

实验 3 对比 SRL 算法和 DST 算法的性能。

考察 SRL 和 DST 算法的性能及约束条件对算法冗余开销的控制, 根据实验 2 的对比结果, SRL 算法选取与 DST 具有相同搜索方向和并发机制的 PAL-RFS 策略。通过改变 PAL-RFS 和 DST 的约束条件, 仿真实验将在资源定位时间和网络带宽耗用两个方面对比不同网络规模下两个算法的性能。模型参数和条件设置同实验 2, 结果如图 4 所示。

通过图 4(a)的对比可知: (1) 由于 DST 中使用 TTL 控制搜索范围是粗粒度的, 故 DST 基本可在 $TTL=1$ 时完成资源定位。TTL 取值不同时, 其带宽耗用情况无明显变化; (2) 相反, PAL-RFS 通过减小搜索半径可有效降低网络带宽耗用; (3) 随着网络规模的增大, DST 算法带

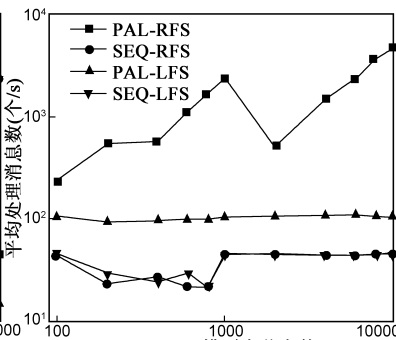
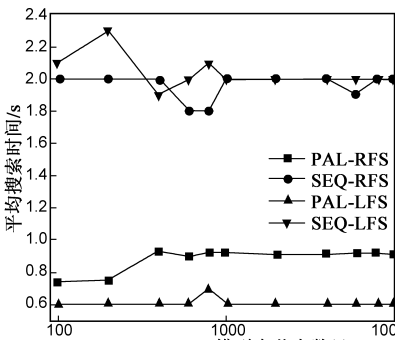


图3 不同网络规模下四种搜索策略的搜索时间和网络带宽使用情况对比

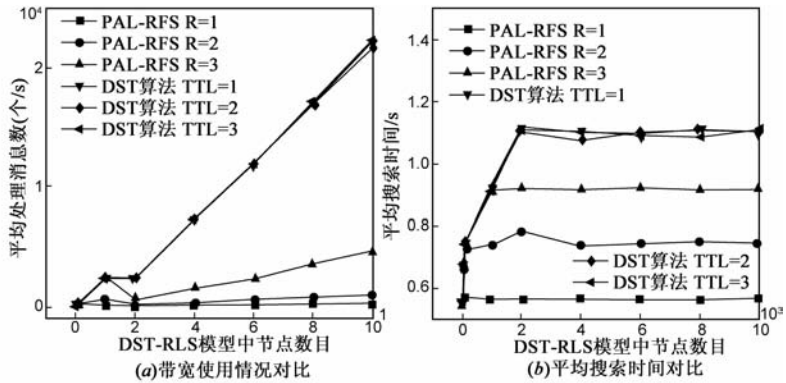


图4 PAL-RFS算法和DST算法的性能对比

宽耗用的增长趋势明显快于 PAL-RFS, 表明 PAL-RFS 的带宽耗用优于 DST。通过图 4(b)对两种定位算法平均搜索时间的对比可知: ①通过限制 TTL, DST 无法有效缩短平均搜索时间; ②由于模型中结点间通信延迟为 200ms, 不同搜索半径时 PAL-RFS 的平均搜索时间相差接近 0.2s; ③通过限制搜索半径, PAL-RFS 的平均搜索时间优于 DST。因此, 在 DST-RLS 模型中, 使用 PAL-RFS 搜索策略的 SRL 算法可通过控制搜索半径, 更加灵活有效地减少带宽耗用和搜索时间, 其性能优于 DST 算法。

5 总结与展望

提出一种基于分布式生成树结构自适应的资源定位服务模型 DST-RLS, 有效解决了云计算中资源定位索引的瓶颈问题。通过有序存放路由表中代表元结点, 给出模型的代表元选取规则, 减小了模型中关键结点出现的概率, 增强了系统的可靠性和负载均衡性。同时, 给出一种搜索半径限制的资源定位算法 SRL, 在逻辑资源索引均匀分布的模型中, 通过选取合适的搜索半径, SRL 算法可在确保成功定位目标资源的同时减少带宽耗用和资源定位时间。算法的时间复杂度是常数阶, 其性能优于同类 DST 算法。

采用 DST-RLS 模型及 SRL 算法, 中小企业可更为便捷地部署自己的云计算环境, 从而保证信息的私有性。若将 DST-RLS 模型中结点的逻辑关系映射到结点的物理位置上, 可通过 NT 技术解决网络拓

扑不匹配问题^[1]。若 SRL 算法选用不同的搜索策略, 可实现对不同地理范围内资源的遍历和使用; 进一步根据用户等级和付费信息, SRL 算法可向不同用户提供差异化服务。DST-RLS 模型和 SRL 算法是对现有云存储模式的一个补充, 在云计算环境中有很好的应用前景。

参考文献:

[1] 陈康, 郑纬民. 云计算: 系统实例与研究现状 [J]. 软件学报, 2009, 20(5): 1337-1348.

- Chen K, Zheng W M. Cloud computing: system instances and current research[J]. Journal of Software, 2009, 20(5): 1337 – 1348. (in Chinese)
- [2] Zhang Y, Zhou Y. 4VP: A novel Meta OS approach for streaming programs in ubiquitous computing[A]. The IEEE 21st International Conference on Advanced Information Networking and Applications (AINA-07) [C]. Niagara Falls, Canada, 2007. 394 – 403.
- [3] Keahey K, Tsugawa M, Matsunaga A, et al. Sky computing [J]. Internet Computing, IEEE, 2009, 13(5): 43 – 51.
- [4] Ghemawat S, Gobioff H, Leung ST. The Google file system [A]. Proc of the 19th ACM Symp on Operating Systems Principles [C]. New York: ACM Press, 2003. 29 – 43.
- [5] Cameron D, Guy L, Kunszt P, et al. Replica management in the european datagrid project [J]. Journal of Grid Computing. 2004, 2: 341 – 351.
- [6] Cai M, Chervenak A, Frank M. A peer-to-peer replica location service based on a distributed hash table[A]. Proceedings of the 2004 ACM/IEEE Conference on Supercomputing [C]. IEEE Computer Society, 2004.
- [7] Xiaohua S, Heqing H, Yuanda C. A scalable evenly distributed replica location mechanism[J]. Digital Media and its Application in Museum & Heritages, Second Workshop on. 2007, 10: 281 – 285.
- [8] Dahan S, Nicod J, Philippe L. The distributed spanning tree: A scalable interconnection topology for efficient and equitable traversal [A]. Cardiff, Wales, United Kingdom: Institute of Electrical and Electronics Engineers Computer Society [C]. Piscataway, NJ 08855-1331, United States, 2005.
- [9] Dahan S, Dobrila A, Nicod J, et al. Performances study of the distributed spanning tree an overlay network for server lookup [A]. Athens, Greece: Institute of Electrical and Electronics Engineers Computer Society [C]. Piscataway, NJ 08855-1331, United States, 2008.
- [10] Lv Qin, Cao Pei, Cohen Edith, et al. Search and replication in unstructured peer-to-peer networks [A]. ACM SIGMETRICS 2002 International Conference on Measurement and Modeling of Computer Systems [C]. Marina Del Rey, CA, United States: Association for Computing Machinery, 2002. 258 – 259.

[11] 雍兴辉, 黄永峰. P2P 流媒体系统中层次化网络[J]. 电子学报, 2010, 38(1): 1 – 5.

Yong Xing-hui, Huang Yong-feng. Network topology hierarchy interface on P2P media streaming system [J]. Acta Electronica Sinica, 2010, 38(1): 1 – 5. (in Chinese)

作者简介:



王铁军 男, 1979年6月生于辽宁锦州. 2005年毕业于电子科技大学, 获硕士学位. 现于电子科技大学计算机科学与工程学院攻读博士学位. 主要研究方向为网络计算、对等网络、信息安全等.
E-mail: wtj. uestc@gmail.com



刘恒 女, 1980年10月生于四川攀枝花. 2005年毕业于电子科技大学, 获硕士学位. 现于电子科技大学计算机科学与工程学院攻读博士学位. 主要研究方向为普通计算、网络计算、信息安全等.
E-mail: noza.liu@gmail.com



孙明 男, 1978年8月生于四川南充. 2009年毕业于电子科技大学, 获博士学位. 现于电子科技大学计算机科学与工程学院工作. 主要研究方向为知识发现、语义 Web 使用挖掘及本体学习.



周明天 男, 1939年3月生于广西容县. 毕业于哈尔滨工业大学, 现为电子科技大学教授, 博士生导师. 主要研究方向为网络计算、分布式计算和信息安全.