

# CACC:一种基于分簇的自适应网络坐标计算方法

李小勇 王意洁

(国防科学技术大学计算机学院并行与分布处理国家重点实验室 长沙 410073)  
(sayingxmu@163.com)

## CACC: An Adaptive Network Coordinate Computing Method Based on Clustering

Li Xiaoyong and Wang Yijie

(National Key Laboratory for Parallel and Distributed Processing, College of Computer, National University of Defense Technology, Changsha 410073)

**Abstract** Aiming at the problem of predicting distances between network nodes using coordinate-based method in the Internet environment, an adaptive network coordinate computing method CACC is put forward based on clustering. Firstly, E-Binning clustering algorithm is proposed, and based on which different strategies of selecting landmarks have been studied. The CACC adopts the optimal distributed landmark strategy, Hybrid. Secondly, an adaptive coordinate updating algorithm ACU is proposed to improve the convergency and accuracy of the coordinate computing. The accuracy of CACC is validated by comparing it to the classic network computing methods GNP and Vivaldi under the relative error and SRRL metrics. The results of the experiment show that CACC is an application-oriented network prediction method that has good scalability, convergency and accuracy.

**Key words** distance prediction; network coordinate; overlay; latency estimation; topology-awareness

**摘要** 针对 Internet 环境中采用基于坐标的方法预测网络节点间的距离问题,提出了一种基于分簇的自适应网络坐标计算方法 CACC. 首先,提出了 E-Binning 分簇算法,并在此基础上通过比较不同的地标选择策略,CACC 采用了最优的分布式地标策略 Hybrid;其次,提出了自适应坐标更新算法 ACU 以提高坐标计算的收敛速度和精确度;最后,采用相对误差和 SRRL 度量方法,通过与经典的网络坐标计算方法 GNP 和 Vivaldi 对比验证了方法的精确性. 实验结果表明,CACC 是一种可扩展性强、收敛性好、预测精度高的面向实际应用的网络距离预测方法.

**关键词** 距离预测;网络坐标;覆盖网;延迟估计;拓扑感知

**中图法分类号** TP393

随着内容分发网络(content distribution network, CDN)和 P2P(peer-to-peer)等技术的发展,近年来涌现出一大批基于拓扑感知的服务和应用,如覆盖网路由<sup>[1]</sup>、应用层组播<sup>[2]</sup>、拓扑聚合<sup>[3]</sup>、临近服务器选择<sup>[4]</sup>等. 然而网络距离(一般指网络延迟,通常以 round-trip time(RTT)表示)信息是实现拓扑感知的基础,同时也是其他基于网络测量的应用,如网络

监控、网络诊断等的基础. 由于 P2P 网络节点规模庞大,动态性强等特点使得实现网络节点间完全距离测量面临极大困难,因此研究者们更加关注于网络距离预测技术的研究.

网络坐标计算作为网络距离预测的主流方式之一目前受到广泛关注. 网络坐标方法最早由 2002 年卡内基·梅隆大学的 Ng 等人提出的 GNP<sup>[5]</sup>开始.

收稿日期:2008-09-16

基金项目:国家“九七三”重点基础研究发展计划基金项目(2005CB321801);国家自然科学基金项目(60873215,60503042);国家自然科学基金创新研究群众科学基金项目(60621003);高等学校全国优秀博士学位论文作者专项基金项目(200141)

GNP 将 Internet 建模为几何空间(如图 1 所示),将 Internet 中的主机节点映射为欧氏空间中的点,通过绝对坐标计算来预测节点间的距离,其坐标计算的实质为一个求解估计误差与测量误差的多维全局最小化问题. 目前很多方法可进行近似求解,在 GNP 中利用 Simplex Downhill 方法<sup>[6]</sup>计算节点的坐标.

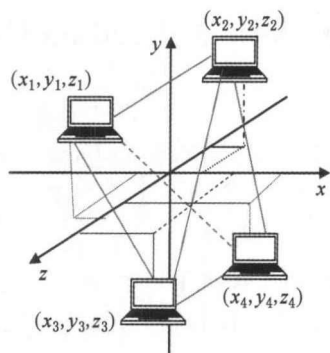


图 1 Internet 的几何空间模型

由于 GNP 采用集中式地标,容易导致通信过载和单点失效,可扩展性不强. 为了消除集中式地标的瓶颈效应,Vivaldi<sup>[7]</sup>采用基于弹簧力场的方式让用户节点以 Piggy-Back 方式通信并且动态地修正网络坐标. Vivaldi 可扩展性强,同时能够及时地反应网络延迟的变化. 然而,Vivaldi 方法不能保证节点能够收敛到稳定状态,并且其收敛速度较慢,以至于在很多情况下不能满足网络应用所需<sup>[8-10]</sup>.

针对已有网络距离预测方法的不足,提出了一种基于分簇的自适应网络坐标计算方法 CACC,实验结果表明该方法具有更好的可扩展性、收敛性以及预测精度等优点,更能够满足实际应用所需.

## 1 CACC 方法模型

CACC 方法采用分布式地标策略,避免了如 GNP 等集中式地标方法使用固定地标所导致的瓶颈、失效等问题. 它结合了 GNP 和 Vivaldi 的优点,同时减少了两种方法所带来的缺陷:其初始坐标计算过程采用 GNP 中使用的绝对坐标方式,并且采用 Simplex Downhill 方法计算坐标,坐标收敛的速度快;与 Vivaldi 方法类似,CACC 中采用分布式地标策略,可扩展性增强,并且通过自适应坐标更新的方式以适应坐标的动态变化. 此外,与上述两种方法均不同的是它使用了多种优化策略:如采用 E-Binning 分簇算法,很多研究已经证明分簇方法能够提高距离预测的精度<sup>[11]</sup>,利用此分簇算法有效地

提高了坐标计算的精度;在分簇的基础上通过比较多种地标选择策略,CACC 采用了最优的地标选择策略 Hybrid;提出自适应坐标更新算法 ACU,以提高坐标的收敛性和计算精度,从而使得该方法更能够满足实际应用所需.

此外,在 CACC 中引入了成员服务器(membership server,MS)来存储系统中所有已计算出坐标的节点信息,包括节点的 IP 地址、节点所属的簇号(即 E-Binning 算法中的 Binning 号)等. 利用 MS 的优点是,通过记录全局信息,为后续的坐标计算过程以及实际的距离预测应用提供帮助. 由于每个节点需要存储的信息量不大,性能较好的服务器便能满足要求. 此外,可采用 DIN<sup>[12-13]</sup>方法来解决因集中式服务器所导致的瓶颈和失效等问题.

在 CACC 中,网络节点的坐标计算主要包括 4 个过程:新加入节点的 E-Binning 分簇过程、新加入节点坐标计算前的地标选择过程、新加入节点的初始坐标计算过程以及节点坐标的自适应更新过程.

在 CACC 方法中,新加入的节点首先通过访问 MS,获得若干个用于分簇的索引节点,并且采用 E-Binning 分簇算法确定该节点所属簇的号数(即 E-Binning 算法中的 Binning 号数)并将相关的信息存入 MS 当中,以便作为其他节点的地标节点;其次,采用最佳的地标选择策略选取一个初始的地标节点集合,为后续的坐标计算做好准备;再次,根据上述过程选取的地标集合,采用非线性优化方法,如 Simplex Downhill 方法计算节点的初始坐标;最后,通过坐标的自适应更新算法 ACU 来更新坐标. 以下各节将详细介绍相关的计算过程和算法.

## 2 CACC 坐标计算方法

### 2.1 E-Binning 分簇算法

CACC 中引入装箱(Binning)策略对新加入的节点分簇. Binning 思想最早由加州大学伯克利分校的 Ratnasamy 等人<sup>[4]</sup>提出,在 Binning 方法的基础上提出了 E-Binning 分簇算法. 该算法的基本思想是两个度量空间中相近的节点具有到其他节点对象的相似距离. 同样地,若两个节点到一组选定的参考节点的距离相似,则认为这两个节点是相近的,因此可将其划分到同一个簇中. 经典的 Lipschitz 嵌入方法<sup>[14-15]</sup>便是基于这种思想,很多利用矩阵分解技术<sup>[14,16]</sup>计算网络坐标的方法都是这种思想的理论基础,其具体为算法 1 所示.

**算法 1. E-Binning 分簇算法.**

输入: 赋予新加入的节点 A, MS 服务器的相关信息, 如 IP 地址等;

输出: 新加入节点 A 的 Binning 号.

① 新加入的节点访问 MS 并获得  $K$  个索引节点信息(如 IP 地址等).

② 对索引节点按序编号, 编号后的索引节点记为  $L_1, L_2, \dots, L_K$ .

③ 节点 A 探测至  $K$  个索引节点的距离, 探测可采用 Ping 的方式, 获得对应的距离信息  $d_1, d_2, \dots, d_k$ , 其中  $d_i$  表示到第  $i$  个索引节点的距离.

④ 将距离进行分级处理, 如 1 级表示延迟在  $[0 \sim 100]$ ms, 2 级表示延迟在  $[100 \sim 200]$ ms, 3 级在  $[200 \sim 300]$ ms, 4 级在  $[300 \sim \infty]$ ms 等. 一般地, 假定将所有的距离范围划分为  $S$  级, 每一级的延迟间隔为  $M$ , 则第  $i$  级的延迟范围为  $[(i-1) \times M, i \times M]$ , 特别地, 第  $S$  级的范围为  $[M \times (S-1), \dots]$ .

⑤ 节点根据所探测的  $K$  个距离信息, 并根据距离级数信息, 确定 Binning 号数. 如到 4 个索引节点的距离分别为 123ms, 55ms, 211ms, 322ms, 那么节点的初始 Binning 号数为 2134.

⑥ 节点访问该 Binning 号所对应的簇中的少量节点(一般为 2~3 个), 探测延迟距离并判断其距离是否在常数  $\delta$ (一般设为较小的数值)之内. 如果是, 则该节点的簇号即为初始 Binning 号, 并在 MS 中存储; 否则转⑦.

⑦ 节点通过 MS 访问扩展的  $K'$ (数量通常较小, 一般为 3 以内)个索引节点, 并采取类似步骤④⑤的方法确定新的 Binning 号数, 如为 ABC, 则该节点最终的 Binning 号为 2134ABC. 在 E-Binning 算法中最终的 Binning 号相同的节点视为相邻节点, 即将其视为在同一个簇中, 并将相关信息存放在 MS 服务器中.

**2.2 地标选择策略**

新节点加入系统后首先访问成员服务器 MS, 通过 MS 获得若干个地标节点信息. MS 中存储的节点信息指系统中所有已计算出坐标的节点信息. CACC 采用分布式地标策略, 即系统中任何已计算出坐标的节点均可成为其他节点的地标, 从而避免了如 GNP 中的集中式地标所带来的瓶颈、失效问题, 极大地提高了系统的可扩展性.

为进一步考察在不同的地标选择策略下网络距离预测的效果, 在分簇的基础上比较了 3 种地标选择策略, CACC 选用最佳的地标节点策略 Hybrid,

提高了坐标计算的精度. 这 3 种地标策略类似于 PIC<sup>[17]</sup>中的 3 种地标选择策略, 但由于 CACC 是建立在 E-Binning 分簇的基础上, 因此在新的环境下考察地标选择策略的影响具有重要的意义. 在 CACC 中 3 种地标节点选择策略描述如下.

1) 随机策略(random): 指在 MS 中随机选择  $K$  个节点作为地标节点, 其中  $K > d$ ,  $d$  为坐标的维数.

2) 临近策略(closest): 新加入的节点通过 E-Binning 分簇方法确定自身所属的簇号(即 Binning 号数), 并且在该簇中选取  $K$  个节点作为地标节点, 若该簇中的节点数小于  $K$ , 则在距其最近的 Binning 中选择地标节点, 其中  $K > d$ ,  $d$  为坐标的维数.

3) 混合策略(hybrid): 结合随机策略和临近策略, 即地标节点的选择尽量分布在各个簇中, 使得地标节点的选择在全局中呈现出较好的分布性.

**2.3 初始坐标计算过程**

在 CACC 方法中, 坐标计算的方式与 GNP 不同, 新加入的节点通过 MS 获得地标信息后连接并访问地标节点, 从而获得地标的坐标信息并且探测到地标的距离信息, 然后将所有相关的数据发送至服务器, 通过服务器来计算坐标并将结果存储在 MS 中以使用户使用. 然而随着网络节点规模的不断扩大, 这种采用集中式服务器计算、通信方式极易造成 MS 负载过重、单点失效等问题, 扩展性不高. 在 CACC 中则采用分布式坐标计算的方式, 坐标的计算过程分布在每一个节点中, 极大地提高了系统的扩展性. 其具体计算过程描述如下:

1) 新加入的节点首先通过查询 MS, 采用 Hybrid 地标策略获得  $K$  ( $K > d$ ,  $d$  为坐标维数)个地标信息;

2) 新加入节点通过探测与询问获得至  $K$  个地标的距离  $d_1, d_2, \dots, d_k$ , 以及  $K$  个地标的坐标信息:  $c_1, c_2, \dots, c_k$ ;

3) 使用 Simplex Downhill 方法计算新加入节

点的坐标, 其目标在于最小化  $\sum_{i=1}^K \left( \frac{d_i - \hat{d}_i}{d_i} \right)^2$ , 其中

$d_i$  为新加入节点和地标  $i$  之间的网络探测距离,  $\hat{d}_i$  为新加入节点与地标  $i$  的坐标空间距离.

**2.4 坐标自适应更新过程**

为适应网络动态环境的变化, 提出了自适应的网络坐标更新算法 ACU (adaptive coordinates updating algorithm), ACU 算法具体描述如算法 2 所示.

算法 2. ACU 坐标更新算法.

输入: 当前节点  $i$  初始坐标  $c_i$ 、各索引节点  $j$  的坐标  $c_j$  以及节点  $i$  至各索引节点的探测距离  $d_{ij}$ , 其中  $j=1, 2, \dots, k, k$  为索引节点的数目,  $Samples$  表示所有的索引节点集合;

输出: 自适应动态调整后当前节点新的坐标值  $C_i^{new}$ .

① 查询各索引节点  $j$  的坐标, 并计算其相对误差  $rerr_j = \frac{|d_{ij} - \|C_i - C_j\||}{\min(C_i, C_j)}$ .

② 计算各索引节点的误差权重  $w_j$  以及总的预测误差  $E_i$ .

$$w_j = \frac{rerr_j}{\sum_{k \in Samples} rerr_k},$$

$$E_i = \sum_{k \in Samples} |d_{ik} - \|C_i - C_k\||^2.$$

③ 计算新的坐标  $C'_i$ , 使得加权后的预测误差  $\sum_{k \in Samples} w_k |d_{ij} - \|C_i - C_k\||^2$  最小化.

④ 检查新得到的预测误差  $E'_i = \sum_{k \in Samples} |d_{ij} - \|C'_i - C_k\||$ .

⑤ 根据  $E'_i$  与  $E_i$  的关系, 调整坐标:  $C_i^{new} = C_i + \epsilon(C'_i - C_i), \epsilon = \min\left(\frac{E'_i}{E_i}, 1\right)$ .

ACU 算法通过引入权重参数并且结合启发式思想自适应地更新坐标. 与 Vivaldi 不同, ACU 坐标更新算法首先通过一次与多个地标节点交互, 并且根据与各地标节点的误差程度, 给予每个地标不同的误差权重因子; 其次, 利用与各地标误差的加权因子重新计算总的加权后的误差; 再次, 通过非线性优化方法最小化总的误差值, 并计算出相应的坐标值; 最后通过误差验证过程来检查新的坐标下总的误差, 并根据新的坐标精度自适应地更新坐标, 从而使得坐标值朝着更精确的方向更新.

3 性能测试

在实验测试中使用的数据集为 MIT King 数据集, 该数据集包含 1740 个 DNS 服务器节点之间的延迟矩阵信息, 节点之间的延迟通过 King 方法收集<sup>[7,18]</sup>, 目前大多数网络坐标的研究都采用此数据集, 因此较具有权威性. 为详细考察 CACC 坐标计算方法的性能, 从不同地标选择策略、坐标的收敛性、基于坐标预测网络距离的精确性 3 个方面进行

实验验证.

3.1 不同地标选择策略对比

通过实验模拟对比了 3 种地标选择策略, 即前面所述的 Random, Closest, Hybrid 策略, 对比结果如图 2 所示. 结果表明 Random 与 Closest 策略结果的精确度比较接近, 而 Hybrid 策略性能最好, 这也表明 Random 策略在地标的全局中的随机分布性, 而 Closest 策略则注重地标选择的局部性, 这两种地标策略其实是两种极端, 易造成远距离与近距离预测精度的不一致性, 而 Hybrid 则综合了它们各自的优点, 实验结果表明采用此地标选择策略效果最佳.

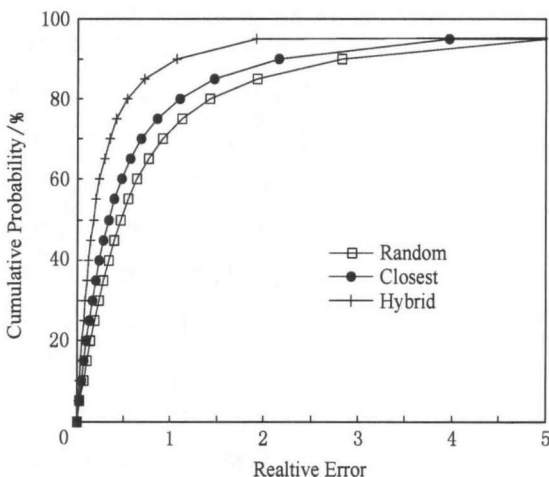


图 2 不同地标选择策略对比

3.2 收敛性对比测试

通过模拟实验对 Vivaldi 和 CACC 坐标计算的收敛性进行了对比测试. 通过对比不同抽样次数(即坐标更新的次数)后坐标的精确性来分析不同方法的收敛速度. 由图 3 可知, CACC 方法通过 30 次抽样后距离预测的精度就已超过 Vivaldi 方法(其中  $d=2, C_c=0.25$ ) 150 次抽样的精度, 而 CACC 方法

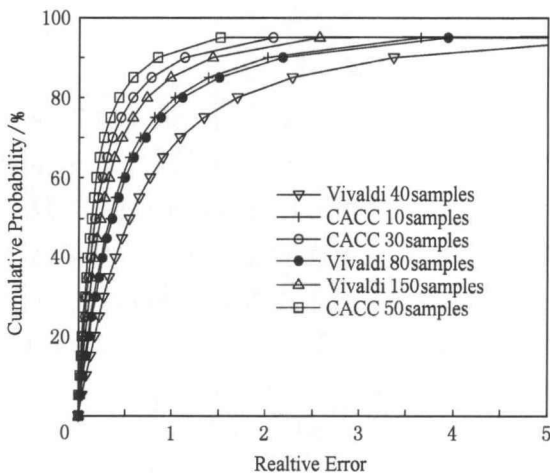


图 3 坐标的收敛性对比

10 次抽样其精度就优于 Vivaldi 方法 80 次抽样后的精度,因此 CACC 采用更少的抽样次数就能达到较高的精度,即欲使二者达到相同的预测精度,CACC 需要的抽样次数更少,因此 CACC 的收敛速度比 Vivaldi 更快。

3.3 CACC 与 GNP, Vivaldi 精确度对比

在模拟实验中,采用两种度量方法来考察 CACC 方法的精确性。

1) 相对误差 (relative error):  $RelError = \|d_{xy} - \hat{d}_{xy}\| / \min(d_{xy}, \hat{d}_{xy})$ , 其中  $d_{xy}$  为节点  $x$  和  $y$  之间的实际探测距离,而  $\hat{d}_{xy}$  为节点  $x$  与  $y$  之间采用坐标计算所得的距离。

2) SRRL<sup>[19]</sup> (smart relative rank loss): 该度量方法是 RRL<sup>[20]</sup> 的扩展形式, RRL 描述了距离某个节点远近次序的缺失,而 SRRL 则在一定程度上容忍相对次序缺失。因为在实际应用中,如果其预测的临近节点与实际节点非常临近,则认为预测依然正确,这种一定程度上的容忍对于真实的网络应用是很有意义的。SRRL 可表示如下:

$$\sigma_{rrl} = \frac{1}{N(N-1)(N-2)} \sum_{i,j,k,i \neq j \neq k} \varphi(d_{ij}, d_{ik}) \cdot 1_{d_{ij} < \hat{d}_{ik}},$$
$$\varphi(d_2, d_1) = 1 - \exp[-\max(0, \frac{d_2 - d_1 - d_{min}}{d_1 + d_{min}})],$$

其中,  $N$  为节点数目,  $d_{ij}$  和  $\hat{d}_{ij}$  分别表示  $i, j$  的测量距离与预测距离,  $d_{min}$  为距离相差的最小阈值。

通过图 4 所示可知,采用相对误差度量方法, CACC 的精确度相对最佳。在经过 60 次更新后,相对误差在 0.5 以内的节点占近 90%,与其他方法相比,精确度明显要高。同时,通过表 1 和图 5 可知,在采用 SRRL 度量方法(实验中  $d_{min}$  设为 10ms)来对比 GNP, Vivaldi 和 CACC 的精确性,发现 CACC 的

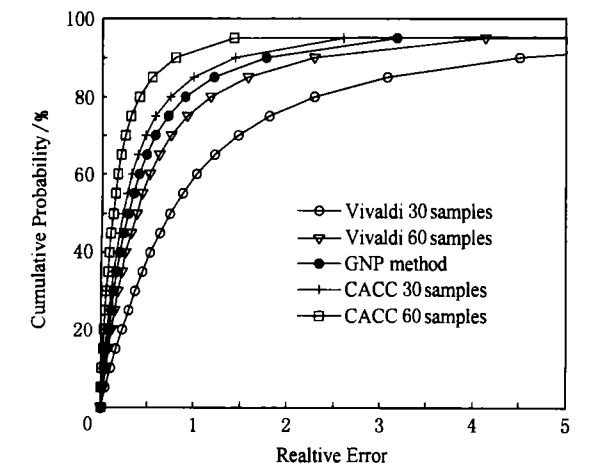


图 4 精确度对比(Relative Error 度量方法)

性能也是最优的。

表 1 SRRL 度量方法下各种方法精确度对比 %

度量方法	SRRL
GNP	9.81
Vivaldi(50 次抽样)	30.43
Vivaldi(100 次抽样)	18.67
Vivaldi(200 次抽样)	9.44
CACC(10 次抽样)	9.62
CACC(30 次抽样)	7.34
CACC(50 次抽样)	4.51

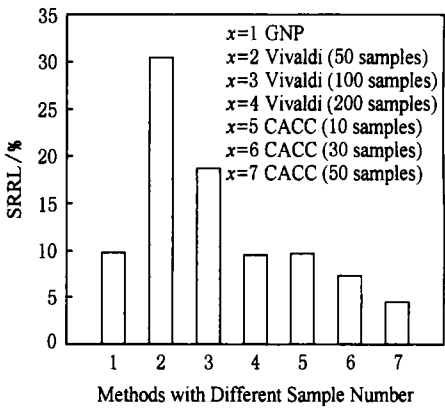


图 5 精确度对比(SRRL 度量方法)

4 结 论

本文提出了一种基于分簇的自适应网络坐标计算方法 CACC,该方法具有以下特点:

- 1) 采用了 E-Binning 有效分簇算法,该算法对已计算出坐标的节点进行分簇,是新加入节点分布式地标选择和某些基于临近性估计应用的基础;
- 2) 在分簇的基础上采用最优的 Hybrid 分布式地标策略,提高了坐标计算的可扩展性和精确性;
- 3) 使用自适应坐标更新算法 ACU 有效地更新坐标,加速了坐标的收敛速度,提高了距离预测精度。

利用相对误差和 SRRL 度量方法,通过与 GNP, Vivaldi 方法的对比,证明了 CACC 具有最佳的精确度。由于以上度量方法以及快速收敛性均面向实际应用,因此该坐标计算方法具有更强的实用性。

参 考 文 献

[1] Rewaskar S, Kaur J. Testing the scalability of overlay routing infrastructures //Proc of the Passive Active Measurement (PAM) Workshop. 2004

- [2] Chu Y, Rao S G, Seshan S, et al. A case for end system multicast. *IEEE Journal on Selected Areas in Communications*, 2002, 20(8): 1456-1471
- [3] Awerbuch B, Shavitt Y. Topology aggregation for directed graphs. *IEEE/ACM Trans on Networking (TON)*, 2001, 9(1): 82-90
- [4] Ratnasamy S, Handley M, Karp R, et al. Topologically-aware overlay construction and server selection //Proc of IEEE INFOCOM, 2002
- [5] Ng T S, Zhang H. Predicting internet network distance with coordinates-based approaches //Proc of IEEE INFOCOM 2002. 2002: 170-179
- [6] nelder J A, mead R. A simplex method for function minimization. *Computer Journal*, 1965, 7(4): 308-313
- [7] Dabek F, Cox R, Kaashoek F, et al. Vivaldi: A decentralized network coordinate system //Proc of ACM SIGCOMM 2004. New York: ACM, 2004: 15-26
- [8] Pietzuch P, Ledlie J, Seltzer M. Supporting network coordinates on planetLab //Proc of WORLDS 2005. 2005: 19-24
- [9] Ledlie J, Pietzuch P, Seltzer M. Stable and accurate network coordinates //Proc of IEEE ICDCS 2006. 2006
- [10] Ledlie J, Gardner P, Seltzer M. Network coordinates in the wild //Proc of NSDI. 2007
- [11] Zhang R, Hu Y C, Lin X, et al. A hierarchical approach to internet distance prediction //Proc of the 26th IEEE Int Conf on Distributed Computing System(ICDCS). 2006
- [12] Lee S J, Banerjee S, Sharma P, et al. Bandwidth-aware routing in overlay networks //Proc of IEEE INFOCOM 2008, 2008
- [13] Fonseca R, Sharma P, Banerjee S, et al. Distributed querying of internet distance information //Proc of IEEE Global Internet Symposium. 2005
- [14] Lim H, Hou J C, Choi C H. Constructing internet coordinate system based on delay measurement //Proc of the 2003 ACM SIGCOMM Conf on Internet measurement. 2003: 129-142
- [15] Tang L, Crovella M. Virtual landmarks for the internet //Proc of the 3rd ACM SIGCOMM Conf on Internet Measurement. 2003: 143-152
- [16] Mao Y, Saul L K, Smith J M. IDES: An internet distance estimation service for large networks. *IEEE Journal on Selected Areas in Communications*, 2006, 24(12): 2273-2284
- [17] Costa M, Castro M, Rowstron R, et al. PIC: Practical Internet coordinates for distance estimation //Proc of the 24th Int Conf on Distributed Computing Systems. 2004: 178-187
- [18] Gummadi K P, Saroiu S, Gribble S D. King: Estimating latency between arbitrary internet end hosts //Proc of the 2nd ACM SIGCOMM Workshop on Internet Measurment. 2002: 5-18
- [19] P key, L m thomson, D t thomson. Non-metric coordinates for predicting network proximity //Proc of the 27th Conf on Computer Communication, 2008
- [20] Lua E K, Griffin T, Pias M, Zheng H, et al. On the accuracy of embeddings for internet coordinate systems //Proc of ACM IMC. 2005

**李小勇** 男,1982年生,硕士研究生,中国计算机学会学生会员,主要研究方向为网络计算、网络测量。

**王意洁** 女,1971年生,教授,博士生导师,中国计算机学会高级会员,主要研究方向为网络计算、数据库技术、移动计算等。