

## 位置感知的覆盖网构建算法\*

张小明,王意洁

(国防科技大学 计算机学院 湖南 长沙 410073)

**摘要** 提出了一个基于底层网络位置信息建立 P2P 覆盖网的算法 Lanet。覆盖网中每个节点都选择和本节点物理距离相近的节点作为自己的邻居,逻辑上相邻的节点也是物理上相近的节点。由于路由路径中每跳的延时都较低,这使得所建立的覆盖网络具有非常低的延时伸展率(latency stretch)。Lanet 不需要网络中固定的节点集充当地标节点,具有很好的分布性和可扩展性。模拟实验结果表明,利用该算法建立的位置感知的 P2P 系统能够大幅度降低数据定位的延时。

**关键词** peer-to-peer;位置感知;覆盖网;网络坐标;数据定位

**中图分类号** TP393 **文献标识码** A

## Research of Location-aware Overlay Construction Algorithm

ZHANG Xiao-ming, WANG Yi-jie

(College of Computer, National Univ. of Defense Technology, Changsha 410073, China)

**Abstract** An algorithm named Lanet whereby the overlay network construction is based on the location information of underlying physical topology is presented. Each peer is connected to a small subset of the other peers which are close in the physical network. The peers that are close in the logical network are also close in the physical network. As a result, the expected delay of each hop in the routing path is very low. This leads to low latency stretch of the overlay network. This algorithm is scalable and distributing for a fixed subset of nodes act as landmarks is not required. Simulation results indicate that the latency of data location in these location-aware systems that are constructed with Lanet can be significantly decreased.

**Key words** peer-to-peer; location-aware; overlay; network coordinate; data location

目前正在使用或研究当中的 P2P (Peer-to-Peer) 网都是在物理网络上建立一层逻辑覆盖网。在这些应用中每个参与的节点都是选择系统中的一部分节点建立逻辑的连接,节点间通过这种连接关系建立一层覆盖网络。但是现在的 P2P 应用中很少考虑逻辑路径与物理路径的一致性问题,这严重影响了路由效率。因此在路由过程中如何充分利用节点的物理位置信息非常重要。

非结构化 P2P 系统通过剪除低效的链接并且选择相近的节点为逻辑邻居来提高覆盖网络中搜索效率<sup>[1]</sup>。结构化 P2P 系统中利用物理位置信息提高路由效率主要有三种方法:短路径路由,近距离邻居的选择和基于拓扑结构的节点 ID 分配。

在短路径路由方法中<sup>[2]</sup>,平均路由延时并不很短。在近距离邻居选择方法中<sup>[3]</sup>,但路由表中随着层次的下降可供选择的节点呈指数形式下降,所以该方法对系统性能的提高有限。在基于拓扑结构的节点 ID 分配方法中<sup>[4]</sup>,节点在加入系统时选择物理距离较近的节点为邻居,但是已有的系统都需要固定的节点充当地标的角色,容易造成小部分节点成为系统的瓶颈,而且覆盖网仍然具有较大的延时伸展率。

本文提出一个基于节点物理坐标构建位置感知的 P2P 覆盖网的算法 Lanet (location-aware overlay network)。Lanet 根据节点的物理坐标来建立逻辑邻居关系,逻辑上也相邻的节点物理上也相邻,同已有的算法相比,该算法具有很好的分布性和可扩展性,不需要大量增加底层网络的开销,构建的覆盖网能

\* 收稿日期:2006-04-03

基金项目:国家 973 重点基础研究发展规划(2002CB312105);高等学校全国优秀博士学位论文作者专项资金项目(200141)

作者简介:张小明(1980—),男,硕士生。

够与底层网络很好的匹配。利用 Lanet 来构建位置感知的结构化和非结构化 P2P 系统,能够大幅度降低 P2P 系统中数据定位的延时。

## 1 节点定位技术

全局网络定位系统(GNP(Global Network Position))<sup>5]</sup>把网络当作一个地理空间模型,但是 GNP 需要固定的节点充当地标,且地标节点需分布在整个网络,远程地标的选取对本地相近的节点间的距离估算没多大意义。实际网络中节点的分布符合幂律模型(power-law),因此节点坐标的计算应当考虑节点的局部位置信息。

纯 P2P 网络定位系统(pure peer-to-peer network position 或 TPNP)对 GNP 进行了改进<sup>6]</sup>。TPNP 的基本思想和 GNP 的思想相似,节点加入到 TPNP 时首先查找已存在于系统中的附近的节点,然后选取查找到的节点中的部分节点作为参考节点,这些参考节点的作用和 GNP 中的地标节点的作用一样,新节点利用这些参考节点的坐标和新节点到参考节点间的延时计算出本节点的坐标。选择那些节点作为新节点的参考节点通过以下 3 个方法得到:

- (1) 取一部分和新节点在同一个域的邻近节点,取一部分和新节点不在一个域的远程节点。
- (2) 取离坐标源节点较近的节点(例如坐标值较小的节点)。
- (3) 取离新节点最近的节点。

第一个方法主要是保证新加入节点的坐标计算不仅收敛于本地,同时在全局也收敛,这个方法考虑了节点的局部位置信息。第二个方法主要是保证系统节点的坐标收敛于一个固定的点,以防止系统中节点坐标进入一种混沌状态。第三个方法主要考虑减少坐标计算对底层网络带来的开销。该算法不需要固定的节点充当地标,充分考虑了网络的局部性,是一个分布式的算法。实验结果表明利用 TPNP 计算出的节点网络坐标的精确度不低于 GNP。

## 2 位置感知的覆盖网构建

### 2.1 覆盖网结构设计

构建位置感知的 P2P 覆盖网应尽量使逻辑网络与底层物理网络匹配,同时所应用的方法又要考虑 P2P 覆盖网的特性,即应具有很好的分布性和可扩展性。Lanet 的思想是:

- (1) 网络被建模成  $n$  维坐标的几何坐标空间模型,坐标空间被网络中的所有节点划分,每个加入的节点都拥有自己的子空间,不同节点拥有的子空间不相交;
- (2) 坐标空间被用来存储节点的物理位置信息,任何节点的坐标都包含在自己所拥有的子空间内,节点坐标的获取不依赖于固定的地标节点集;
- (3) 每个节点都有一张邻居节点列表,列表项保存子空间和与自己子空间相邻的节点的 IP 地址与子空间,邻居节点列表被节点加入时当作路由表来使用,系统中的节点通过这种邻居关系建立连接并组织成覆盖网;
- (4) 邻居关系的定义为:在  $d$  维空间中如果两个子空间有  $d-1$  维重叠,有一维相邻接,则这两个子空间相邻,如果两个节点所在的子空间相邻,则这两个节点相邻。

### 2.2 覆盖网构建算法 Lanet

由于坐标空间被系统中的所有节点划分,因此当新节点要加入到覆盖网中时,它需要拥有自己的坐标值和子空间。覆盖网中节点坐标值是通过文献[6]中改进的 GNP 算法 TPNP 计算得到的,即新节点选择一部分物理距离邻近的节点和一部分坐标值较小的远程节点作为参考节点计算出自己的坐标值。新节点的子空间通过划分系统中已有节点的子空间得到。节点加入过程算法的源语描述如图 1 所示, $A, B, C$  都表示节点,其中  $A$  表示新节点,Set,  $s_{land}$  表示节点集。Lanet 的理论通信开销为  $O(d/4 \times n^{1/d})$ ,然而在实际中,由于新节点都是经由附近的节点加入到系统中,因此实际的通信开销要比理论值小很多,只有  $O(d)$ 。

节点子空间的划分过程为:假设目标节点拥有的子空间为 $(a_1 - b_1, a_2 - b_2, \dots, a_n - b_n)$ ,坐标为 $(x_1, x_2, \dots, x_n)$ 。新加入的节点坐标值为 $(y_1, y_2, \dots, y_n)$ ,则选择使坐标差值 $|x_i - y_i| (1 \leq i \leq n)$ 值最大的维进行划分,假设在第 $h$ 维上坐标差值最大,划分给新节点的坐标空间为 $(c_1 - d_1, c_2 - d_2, \dots, c_n - d_n)$ ,则如果 $y_h > x_h$ , $c_h = (x_h + y_h) / 2$ , $d_h = b_h$ , $b_h = c_h$ ;否则 $c_h = a_h$ , $a_h = (x_h + y_h) / 2$ , $d_h = a_h$ ;其中 $c_k = a_k$ , $d_k = b_k (0 < k \leq n$ 且 $k \neq h)$ 。

1. A 寻找到系统中一个节点 B;
2. 从 B 中获取参考节点集 Set;
3. 从 Set 中选取  $n$  个参考节点  $s\_land$ ;
4. 计算出 A 的坐标值 A.Cordinate;
5. 利用 A.Cordinate 从 B 开始路由找到目标节点 C,C 的子空间 C.Space 包含 A.Cordinate;
6. A 的子空间 A.Space\_c = 从 C.Space\_c 中划分出的一部分;
7. A 的邻居列表 A.neighborList\_c = C 的邻居列表 C.neighborList\_c  $\cup$  {C};
8. A 和邻居节点建立邻居关系并修改 A.neighborList\_c;
9. C 和邻居节点都修改 neighborList\_c;

图 1 节点的加入过程

Fig.1 Process of node joining

### 3.1 位置感知的 CAN

CAN 中的节点是建立在虚拟的  $d$  维笛卡儿空间中,该空间完全是逻辑上的,和底层网络没有任何联系。在 CAN 的构建机制中,节点加入时是随机的选择一个区域,因此节点的邻居在物理上并不一定相近,这就引起了路由的低效率问题,因为网络中的任何节点都可能成为路由中的下一步。有些研究也提到一些改进算法来降低 CAN 路由的延时,这些算法都是试图在已有的路径中选择一条较好的路径,我们在这里将彻底改变 CAN 逻辑网结构。

利用 Lanet 算法来构建一个位置感知的 CAN(LCAN),系统中需要两个空间:一个物理坐标空间和一个虚拟的笛卡儿空间,虚拟空间的大小根据 CAN 中用户发布的数据的关键字 Hash 值的范围而定。物理坐标空间用于保存节点的物理位置信息,虚拟空间用于存储(关键字,值)(key,value)对。坐标空间和虚拟空间都被系统中所有的节点划分,每个节点保存两个子空间,一个物理子空间和一个虚拟子空间,物理子空间包含该节点的坐标值,虚拟子空间包含该节点的存储的关键字的 Hash 值,新节点的虚拟子空间从目标节点虚拟子空间划分出一半得到,目标节点即为物理子空间包含新节点坐标值的节点。节点同时维护两份邻居节点列表,一个物理邻居列表保存物理上与本节点相邻的节点的信息,另一份逻辑邻居列表保存虚拟空间中与本节点相邻的节点的信息。物理空间邻居节点列表在系统建立时使用,虚拟空间邻居节点列表在数据查找时使用。虚拟空间中的邻居关系的定义和 2.1 节中邻居关系的定义一样。LCAN 的建立过程的算法描述如图 2 所示,A,B,C 都表示节点,其中 A 表示新节点,Set,s\_land 表示节点集。在 LCAN 中,虚拟空间中相邻的节点也是物理上比较相近的节点,即逻辑上相邻的节点也是物理上相邻的节点。

### 3.2 位置感知的 Gnutella

在非结构化 P2P 中,每个节点都是随机的选择邻居节点。如果节点在选择邻居节点时能够知道其他节点的物理位置信息,则节点能够选择物理上与自己相对较近的节点作为自己的邻居,这将有利于逻辑网络中路由的平均延时的降低,从而能够减少底层网络的负载。但是即使每个节点知道全局中任何节点间的相对位置,也不能构造一个最优的覆盖网,因为这是一个 NP 问题。

可以利用 Lane 算法来建立位置感知的 Gnutella(LGnutella),采用一个启发式过程来选择 Gnutella 节点的邻居<sup>[4]</sup>。假设每个节点拥有  $k$  个邻居,则选择  $k/2$  个物理距离相近的节点,另外  $k/2$  个邻居节点从系统中随机选取。首先利用 Lanet 算法建立节点的邻居关系,然后从节点的邻居节点列表中选取  $k/2$  个最近的节点,如果邻居节点列表的大小小于  $k/2$ ,则从最近的邻居节点的邻居列表中选取  $k/2$  中不

足的部分节点,如果还不满则从次近的邻居节点的邻居列表选取,直到找到  $k/2$  个距离相近的节点。每个节点都用一张列表来保存系统中远程节点的信息,该表采用最近最少使用替换策略维护,每当节点转发数据查找和发布消息时,都用消息的源节点替换掉该表中最近最少使用的节点。 $k/2$  个随机的邻居节点就从这张保存远程节点信息的列表选取。

选择  $k/2$  个物理上较近的节点作为邻居主要是减少路由每步的平均延时,选择  $k/2$  个随机的节点作为邻居主要是为了加强本地节点与外界的联系,增强系统的稳定性,同时有利于提高查找的成功率。

1. A 寻找到系统中一个节点 B ;
2. 从 B 中获取参考节点集 Set ;
3. 从 Set 中选取  $n$  个参考节点  $s\_land$  ;
4. 计算出 A 的坐标值 A.Cordinate ;
5. 利用 A.Cordinate 从 B 路由找到目标节点 C ,C 的物理子空间 C.Space 包含 A.Cordinate ;
6. A 的物理子空间 A.Space\_c = 从 C.Space\_c 中划分出的一部分 ;
7. A 的虚拟子空间 A.Space\_v = 从 C 的虚拟子空间 C.Space\_v 中划出的一半 ;
8. A 的物理邻居列表 A.neighborList\_c = C 的物理邻居列表 C.neighborList\_c  $\cup$  {C} ;
9. A 的逻辑邻居列表 A.neighborList\_v = C 的逻辑邻居列表 C.neighborList\_v  $\cup$  {C} ;
10. A 和物理邻居节点建立邻居关系并修改 A.neighborList\_c ;
11. A 和逻辑邻居节点建立邻居关系并修改 A.neighborList\_v ;
12. C 和物理邻居节点都修改 neighborList\_c ;
13. C 和逻辑邻居节点都修改 neighborList\_v ;

图 2 位置感知 CAN 的构建

Fig.2 Construction of location-aware CAN

## 4 性能分析

通过模拟实验来分析通过建立位置感知的覆盖网对 P2P 数据定位所带来的性能改进。模拟试验中构建网络时节点间的延时从以下两个途径得到 :

(1) P2PSim :P2PSim 项目测量了大约 2000 个网络 DNS 之间的 RTTs 矩阵。

(2) BRITE :BRITE 是一个代表性的网络拓扑生成器,它产生的拓扑图更加接近实际网络。我们用 BRITE 作为产生两层的网络拓扑图。

模拟实验主要验证位置感知覆盖网的延时伸展率和数据定位的延时伸展率,覆盖网的延时伸展率为位置感知的覆盖网中节点间平均延时与底层网络节点间平均延时之比,数据定位的延时伸展率为位置感知 P2P 中系统据定位的平均延时与传统 P2P 系统中数据定位平均延时之比。

### 1. 覆盖网的延时伸展率的测量

利用算法 Lanet 构建的 P2P 覆盖网的延时伸展率的验证,在该实验中覆盖网坐标空间为 7 维。图 3 表现了位置感知覆盖网节点间平均延时伸展率的累加概率的分布情况。从模拟实验结果可以看出,建立的 Lanet 网 70% 以上的节点间的延时伸展率在 2.0 以下,这是由于覆盖网中节点的逻辑邻居也是物理上相近的节点,逻辑网与覆盖网能够很好的匹配。

### 2. LKAN 中数据定位的延时伸展率的测量

第一个实验中构建的 LKAN 的物理空间和虚拟空间都为 7 维。在模拟器中执行 1000 次数据定位请求,图 4 为 LKAN 中平均数据定位延时伸展率的累加概率分布图。可以看出 LKAN 中大部分的数据定位所需的延时不足原始 CAN 中数据定位所需延时的 30% ,这是因为在 LKAN 中节点的邻居都是物理距离相近的节点,在数据定位的消息路由过程中,每个节点转发消息给邻居节点也就是物理距离相近的节点而不是随机的节点,因此消息路由所需的总延时也就相应减少。第二个实验采用 BRITE 产生的 5000 个节点构建不同维数虚拟空间的 LKAN。图 5 为不同维数 LKAN 平均数据定位延时伸展率,由图可以得出 LKAN 的数据定位延时伸展率不受系统虚拟空间维数的影响。

### 3. LGnutella 中数据定位延时伸展率的测量

实验中构建的 LGnutella 的覆盖网坐标空间采用 7 维。模拟器中执行 1000 次数据定位请求,图 6 为 LGnutella 数据定位的延时伸展率累加概率的分布图,结果表明位置感知的 Gnutella 能够大幅度降低数据定位的延时,这是由于 LGnutella 中每个节点都有数个物理距离相近的邻居,这就使得在数据定位的消息路由的路径中,有一部分路由跳点之间的延时即为物理距离相近的节点之间的延时,这些路由跳点的选择减少了路由总延时。

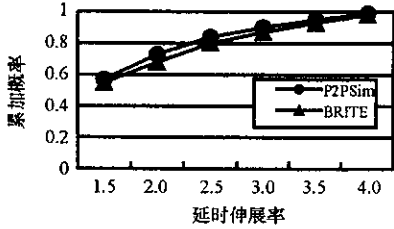


图 3 位置感知覆盖网的延时伸展率  
Fig. 3 Latency-stretch of location-aware overlay

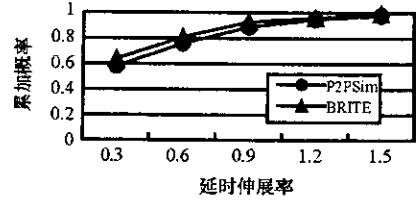


图 4 LCN 的延时伸展率  
Fig. 4 Latency-stretch of LCN

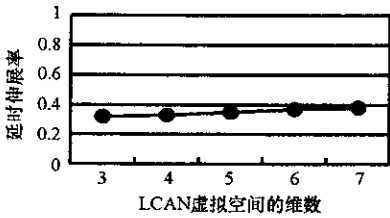


图 5 不同维 LCN 的延时伸展率  
g. 5 Latency-stretch of LCN with different dimensions

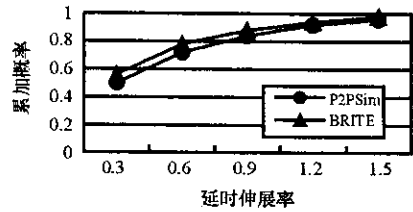


图 6 LGnutella 的延时伸展率  
Fig. 6 Latency-stretch of LGnutella

## 5 结论

本文提出了一个构建位置感知的 P2P 覆盖网的分布式算法 Lanet,利用该算法构建位置感知的覆盖网不需要大量增加底层网络的开销。该算法可以用来构建位置感知的结构化的 P2P 和非结构化的 P2P 系统。我们通过模拟实验测试了 Lanet 的效率,模拟试验中底层网络数据既有来源于真实的网络也有来源于网络模拟器产生的网络,实验结果并没有受到实验环境的影响,结果表明:利用该算法构建的覆盖网节点间逻辑路径的延时与底层物理网络最短路径延时的比值的平均值不到 2.0 倍,同原始相应的 P2P 系统相比,利用该算法构建的位置感知的 P2P 系统能够大幅度降低数据定位的延时。

## 参考文献:

[ 1 ] Liu Y , Xiao L , Liu X , et al . Location Awareness in Unstructured Peer-to-peer Systems[ J ] . IEEE Transaction on Parallel and Distributed Systems , 2005 , 16 : 163 - 174 .

[ 2 ] Castro M , Druschel P , Hu Y C , et al . Topology-aware Routing in Structured Peer-to-peer Overlay Networks[ R ] . Tech. Rep. MSRTR - 2002 - 82 , Microsoft Research , One Microsoft Way , Redmond , WA 98052 , 2002 .

[ 3 ] Winter R , Zahn T , Schiller J . Topology-aware Overlay Construction in Dynamic Network[ A ] . Proceeding 3<sup>rd</sup> International Network[ C ] , 2004 .

[ 4 ] Ratnasamy S , Handley M , Karp R , et al . Topologically-aware Overlay Construction and Server Selection[ A ] . In Proc. 21<sup>st</sup> IEEE INFOCOM[ C ] , New York , June 2002 .

[ 5 ] Ng T S E , Zhang H . Predicting Internet Network Distance with Coordinates-based Approaches[ A ] . In Proceedings of IEEE INFOCOM [ C ] , 2002 : 170 - 179 .

[ 6 ] Onbilger O K , Chen S , Chow R . A Peer-to-peer Network Position Architectur[ R ] . ICON , Gainesville , FL 32611 , 2004 .



