基于混合P2P网络模型的语义检索方法研究

刘 震 邓 苏 黄宏斌
(国防科技大学信息系统与管理学院 C4ISR 重点实验室 长沙 410073)

摘 要 在语义理解的基础上检索出满足用户需求的信息,是P2P走向更广泛应用的关键技术之一。提出了一种支持语义的混合P2P网络模型M-Chord,采用基于元数据规范模板(MST)的语义描述模型,结合Chord和语义覆盖网的技术特点,对基于MST的语义覆盖网动态生成方法进行了设计,提出了语义扩展路由的概念,并在上述研究的基础上提出了语义检索方法。通过实验分析表明,M-Chord具有较好的扩展性和语义检索性能。

关键词 P2P,元数据,语义检索,语义扩展路由
中图法分类号 TP393.02 **文献标识码** A

Research on Semantic Query in Hybrid P2P Networks

LIU Zhen DENG Su HUANG Hong-bin
(College of Information System and Management, National University of Defense Technology, Changsha 410073, China)

Abstract Supporting semantic query is one of key techniques which broaden P2P systems' applications. A semantic-supported hybrid P2P network model—M-Chord was proposed. It adopts a metadata-specification-template-based semantic description model and combines technical characteristics of Chord and semantic overlay. A MST-based semantic overlay construction approach was designed. In M-Chord, the concept of semantic query routing was proposed. Based on M-Chord network model, semantic query method was proposed. The experiments show that scalability and semantic search efficiency are improved greatly in M-Chord.

Keywords P2P, Metadata, Semantic query, Semantic extend routing

1 引言

P2P[1,2]系统是目前分布式系统研究的热点,主要解决分布式环境中的资源定位问题,即在没有集中控制的情况下如何将索引信息分布到各站点上,并利用这些索引信息检索满足特定需求的信息资源。传统的P2P系统多集中在基于关键字(如文件名)的信息检索方面。近年来,随着知识需求的凸显以及网络的发展,语义信息在计算机与信息系统的相关研究中开始呈现越来越重要的作用,人们对信息的需求逐步转化为对知识的需求,这就对P2P系统中信息检索提出了新的挑战,要求提供一种语义检索能力。

语义检索,目前并没有一个确切的定义,它是与传统关键词检索相区别的,是对检索条件、信息组织以及检索结果显示赋予了一定语义成分的检索方式。国内外在支持语义的P2P系统研究方面已开展了大量工作,如欧盟资助的SWAP项目[3]致力于将P2P和语义网结合应用于分布式知识管理,克服了现行的P2P只能通过关键词查询的局限性,取而代之的是在语义层次更为有效和精确的知识共享技术。Castano设计了一个语义P2P网络环境下基于本体的知识发现、知识共享和社区自组织通用型系统框架Helios[4]。文献[5]在非结构P2P系统结构上引入语义相似度,开发了语义化的信息检索系统PeerIS。华中科技大学金海研究了基于语义网格的语义关联存储模型及管理和通信的问题[6],并开发了满足文献元数据共享的系统SemreX,SemreX系统[7]根据节点之间的语义相似度生成语义拓扑网络并提供语义检索能力。这些研究主要采用的是基于扩散的非结构化网络拓扑,支持多样化查询但不易扩展。文献[8]提出的混合结构化和非结构化拓扑的pGroup体系提高了搜索效率和扩展性。文献[9]提出的基于结构化P2P网络路由的语义覆盖网络结构SSON,采用结构化P2P网络中标识符映射机制及路由查找机制,根据资源类别将节点组织成层次化的覆盖网络。在这些方法中,虽然采用了结构化网络来索引语义类别,但类别信息需要在索引前确定,限制了语义P2P网络中Peer节点知识的动态更新。

本文针对上述问题,提出一种支持语义的混合P2P网络模型,在此基础上对语义检索方法进行设计;第2节首先对语义描述模型进行形式化定义;第3节在语义描述模型基础上,对混合P2P网络模型进行设计,介绍了基于扩展Chord[10]协议的语义对象索引方法和基于索引的语义覆盖网动态生成方法,同时为了提高查全率,提出了语义扩展路由的概念;第4节在混合P2P网络模型的基础上对语义检索算法进行了设计;第5节通过模拟试验对系统性能进行了分析;最

到稿日期:2009-03-11 返修日期:2009-06-01
刘 震 博士,讲师,主要研究领域为语义信息管理、P2P、智能决策支持技术,E-mail:hero12251976@163.com;邓 苏 教授,博士生导师,主要研究领域为信息管理、智能决策支持技术;黄宏斌 副教授,主要研究领域为信息管理、P2P。

后进行了总结。

2 语义描述模型

所谓语义信息，它属于知识的范畴，是指与某一研究领域有关的语义实体以及语义实体之间的语义关系。其中的语义关系，揭示了语义实体之间的数量、时间、因果、方式、状态等关系。一般情况下采用本体来规范信息系统中各种信息资源的语义描述。采用一个全局本体，能够有效地保证语义上的一致，但在分布、开放的大规模语义系统环境中，要形成这种大而全的全局本体是不太现实的，因而必然存在多个领域本体对相关领域的语义描述进行规范，同时提供本体间的继承关系。

基于上述考虑，本文提出一种基于元数据模板的语义描述模型，该模型是后面研究的基础。每个元数据规范模板(Metadata Specification Template，MST)建立在相关领域理解的基础上，相当于相关领域的领域本体，MST 的每个实例就是具体信息资源对象的元数据。整体语义描述模型关系结构如图 1 所示。元数据结构的设计是面向特定领域、特定任务或特定团体兴趣域的，它可以在已有的元数据规范的基础上进行扩展，最终生成领域扩展元数据规范模板(Extensional Domain Metadata Specification Template，EDMST)。领域语义词典用来规范模板中元素命名，并定义了不同词汇间的语义关系，映射规则库定义了不同模板元素间的显式映射规则。

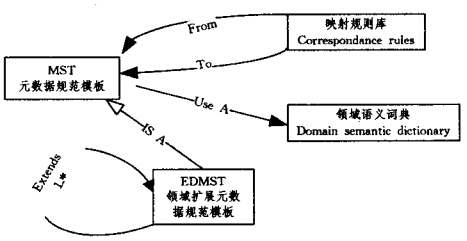

图 1 语义描述模型关系结构

元数据规范模板形式化定义如下：

定义 1 元数据规范模板的抽象模型 $M_{MST}=(MID, Root, S^{IU}, H^{IU}, R^{IU}, P)$。其中，

1)MID 是 MST 的标识，用来唯一区分一个 MST；

2)$Root$ 是 MST 的根元素，$Root\in S^{CIU}$，这是访问 MST 中信息单元的入口；

3)S^{IU} 是 MST 中定义的信息单元集合，$S^{IU}=S^{CIU}\cup S^{BIU}$，$S^{CIU}$ 是复合信息单元集合，S^{BIU} 是基本信息单元集合。$\forall c\in S^{BIU}$，$c=(Name(c), dt)$，$Name(c)$ 表示 c 的命名词汇，$Name(c)\in T$，$dt\in D_{basic}\cup D_{enum}$；$\forall c\in S^{CIU}$，$c=(Name(c), A^{IU})$，$Name(c)\in T$，$A^{IU}=\{x\mid \forall x\in S^{IU}, \exists r(c,x)\in R^{IU}\}$；

4)H^{IU} 是信息单元间的一种层次关系，包括了信息单元之间的 is-a 关系和 part-of 关系，H^{IU} 是一种偏序关系。$H^{IU}\subset S^{IU}\times S^{IU}$，$H^{IU}(c_1, c_2)$，$c_1, c_2\in S^{IU}$，表示 c_2 is-a c_1 或者 c_2 part-of c_1；

5)R^{IU} 是信息单元之间的二元语义关系集合。$\forall r\in R^{IU}$ 可表示为 $N^r(c_1, c_2)$，$N^r\in T$ 为关系的名称，$c_1\in S^{CIU}$，$c_2\in S^{IU}$。如果 $c_2\in S^{CIU}$，则 r 为对象关系；如果 $c_2\in S^{BIU}$，则 r 为属性关系。因而 $R^{IU}=OR^{IU}\cup AR^{IU}$，$OR^{IU}$ 为对象关系集合，AR^{IU} 为属性关系集合；

6)P 是 MST 中所有信息单元和信息单元关系要满足的约束的集合，包括取值约束和基数约束。

MST 虽然采用相同元数据抽象模型来描述，但是这些组织可能采用不同的信息单元来描述相同的事物。面向这一问题，本文提出了 MST 映射规则(Mapping Rules，MR)的概念。

定义 2 MST 映射规则定义了不同 MST 中信息单元之间的对应关系。MR 可以形式化地表示为如下形式：$MR^{MST1\to MST2}: MST1: IU_A \Leftrightarrow MST2: IU_B$。对应关系“⇔”包括等价关系“≻≺”、包含关系“≻”、重叠关系“～”。

$MST1: IU_A \succ\prec MST2: IU_B$，当且仅当两个信息单元的内涵定义是相同的；

$MST1: IU_A \succ MST2: IU_B$，当且仅当信息单元 $MST1: IU_A$ 的内涵定义包含 $MST2: IU_B$ 的内涵定义；

$MST1: IU_A \sim MST2: IU_B$，当且仅当两个信息单元的内涵定义的交集不为空。

MST 的实例化过程实质上是信息资源的元数据描述过程，包括静态实例和动态实例。静态实例是直接定义 MST 中信息单元与具体资源实例值之间的关系，而动态实例是通过映射的方式定义信息单元与具体资源实例之间的关系，如针对数据库资源包括 Global as view，Local as view 和 Both as view 方法。

3 混合 P2P 网络模型

3.1 M-Chord 网络组织模型

信息资源的元数据描述具有如下特点：1)任意信息资源都是依据某一特定 MST 定义的；2)MST 描述的基本单位是语义信息单元，语义检索中的基本组成也是语义信息单元(可能参照某一 MST 定义，或依据元数据抽象模型全新定义)；3)不同 MST 中定义的信息单元存在语义相关性。

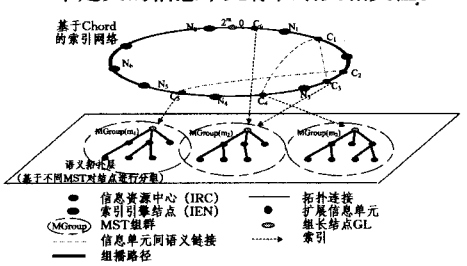

图 2 基于语义的混合 P2P 网络模型 M-Chord

针对这些特点，本文提出了一种基于语义的混合 P2P 网络模型 M-Chord，其总体结构如图 2 所示。总体思想是：1)基于语义信息单元对分布在各个信息资源中心节点(IRC)上的信息资源进行组织索引。本文采用基于 Chord 扩展的 DHT 方式来进行组织索引，利用 DHT 的高效性快速定位包含特定语义信息单元定义词汇的 MST 实例所在的 IRC 或 MST 组群，同时利用这些索引信息动态生成 MST 组群。2)任意信息资源都是依据某一特定 MST 定义的，因而每个 MST 反映了其资源实例的信息能力。当任意语义查询与某一 MST 的语义相似度达到一定程度时，说明包含基于该 MST 描述资源实例的 IRC 具有较大回答相应查询的可能。为了保证查全率，需要在这些 IRC 中传播发现请求，在 M-Chord 中通过将这些包含相同 MST 资源实例的 IRC 组成一个组群，通过构建组播树来提高组内资源发现请求的传播性能，同时通过组播树的根节点来筛选资源发现请求，确定是否需要向所有组成员传播查询，从而可以有效地减轻网络负载。3)不同

MST 中定义的信息单元之间可能存在语义相关性，在 M-Chord 中利用不同信息单元之间存在的语义链接，提供一种面向语义的查询路由扩展能力来提高资源的查全率。

在 M-Chord 中，IRC 可以动态地加入和退出，资源状态也在不断地变化，由此带来资源信息的更新及传播。为了简化问题，便于表述，首先做如下假设。

假设 1　一个资源只有一个信息注册 IRC 节点，该 IRC 上的注册资源被视为其本地资源。

假设 1 主要是为了表述的方便。若一个资源同时在多个 IRC 注册，则其中一个 IRC 将被视为该资源的注册节点，而其它的节点则被视为该资源信息的复制节点，其位置通过注册节点得知，复制问题在本文中暂时不做重点考虑。

假设 2　IRC 覆盖拓扑在一次资源发现过程或一次资源信息更新过程中的变化可以忽略不计，且资源信息在一次资源发现过程中保持有效和稳定。

相对于整个拓扑的演化过程而言，一次资源发现或信息更新的时间是非常短的，因此从网络拓扑演化的角度而言，这样的简化是合理的。

针对以上假设，M-Chord 网络资源信息组织模型描述如下。

定义 3　M-Chord 网络资源信息组织模型 $GM=(R, N^{IRC}, S^{MST}, E, rf)$。其中，

R：M-Chord 网络中的所有注册资源集合；

N^{IRC}：M-Chord 网络中的所有 IRC 节点集合；

S^{MST}：M-Chord 网络中的所有 MST 集合；

E：代表 IRC 间的邻接关系，反映 IRC 的覆盖拓扑结构，$E(n)$为节点 n 的邻节点集合，$n\in N^{IRC}$，$E(n)\subseteq N^{IRC}$；

rf：资源注册函数，$rf: R\times S^{MST}\rightarrow N^{IRC}$，在假设 1 下，$rf$ 为满射。

为方便表示，进一步引入如下定义：

$LMR(n,m)$：结点 n 上基于元数据规范模板 m 注册的本地资源集合，$n\in N^{IRC}$，$m\in S^{MST}$，$LMR(n,m)=\{r\mid rf(r,m)=n\}$。$LMR(n,m)\subseteq R$，$R=\bigcup_{n\in S^{IRC},m\in S^{MST}} LMR(n,m)$。对于任意 $n_1,n_2\in N^{IRC}$，$n_1\neq n_2$，$LMR(n_1,m)\cap LMR(n_2,m)=\Phi$。

$MIE(m)$：元数据规范模板 m 中的可索引的语义信息单元集合，$m\in S^{MST}$。

N^{IEN}：M-Chord 中所有 IEN 节点集合，$N^{IEN}\subseteq N^{IRC}$。IEN 是索引引擎节点(Indexing Engine Node，IEN)，它实质上是具有信息资源索引功能的 IRC 节点，基于 Chord 协议构成一个一维环形网络，利用语义信息单元的定义词汇对分布在不同 IRC 上的 MST 和 MST 组群进行索引。

3.2　IEN 索引结构

IEN 是从 IRC 节点中选取出来的，一般是具有较高级别(隶属单位级别较高)、较长的在线时间、稳定性较高和较大负载能力的 IRC 节点。利用 Chord 协议在 IEN 间维护一个一维环状结构，对于 $\forall n\in N^{IEN}$，都赋予一个 m 位的标识 $id_m(n)$。n 索引内容包括两类，分别为：1)对于任意语义信息单元 c，$name(c)$为 c 的命名词汇，$k=1\oplus id_{m-1}(c)$，其中 $id_{m-1}(c)=\text{hash}_{EIU}(name(c))$；2)对于信息单元 c'，$value(c')$为 c'的静态实例值，$k=0\oplus id_i(c')\oplus id_{m-i-1}(value(c'))$，其中 $id_i(c')=\text{hash}_{CIU}(name(c'))$，$id_{m-i-1}(value(c'))=\text{hash}_{value}(value(c'))$。

通过将 $id_m(n)$和键值 k 对 2^m 取模后顺序排列，n 和键值 k 对应的索引内容被映射到一个大小为 2^m 的环状逻辑空间中。在这个环状空间中，IEN 标识之间、IEN 标识与索引键值之间形成了前驱与后继的关系。Chord 环上若 $id_m(n)$与 k 相等或按顺时钟方向紧随其后，则称 n 为键值 k 的后继节点，表示为 successor(k)。Chord 环上若 $id_m(n)$按逆时钟方向紧随 k 之后，则称 n 为键值 k 的前驱节点，表示为 predecessor(k)。关于键值 k 的索引信息存储到其对应在环状空间中 k 的后继结点 successor(k)上。每个 IEN 包含的索引信息结构如图 3 所示。

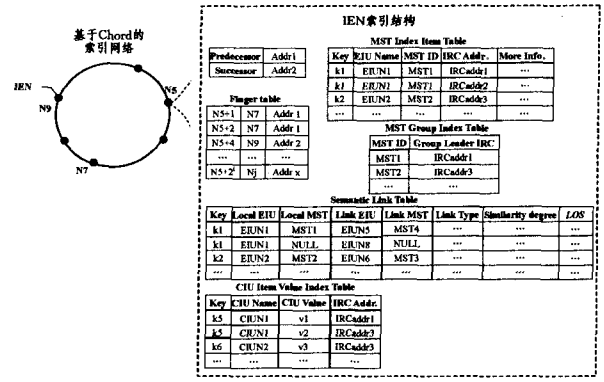

图 3　IEN 的索引信息结构图

与传统 Chord 索引结构类似，为确保高效的路由与查找性能，每个 IEN 除了维护前驱(predecessor)与后继节点(successor)之外，还维护一个名为 finger table 的路由表，其中最多包含 m 个表项，每一个表项都记录着 successor($n+2^{i-1}$)的地址($1\leqslant i\leqslant m$)。同时，为了 M-Chord 组织索引的需要，对传统 Chord 的索引结构进行了扩充，针对每个 IEN 增加了 MST 语义信息单元索引项表(Index Item Table，IIT)、MST 组群索引表(MST Group Index Table，GIT)、语义链接表(Semantic Link Table，SLT)、核心信息单元值索引表(CIU Item Value Index Table，VIT)。

通过 IIT，基于语义信息单元(EIU)名称可以索引到包含特定 EIU 的 MST 所在的 IRC 地址，其中：1)key 对应相应语义信息单元的 Hash 键值；2)EIU Name 对应扩展信息单元名称；3)MST ID 对应扩展信息单元所属 MST 的唯一标识；4)IRC Addr. 对应包含特定 MST 的 EIU 实例的 IRC 地址；5)More Info. 提供了 IRC 结点的相关状态信息。

GIT 包含了 MST 组群的组长节点(Group Leader，GL)的地址信息，其中：1)MST ID 表示 MST 的唯一标识；2)Group Leader IRC 对应 MST 组群的组长节点地址。这些信息是动态生成的。

SLT 记录了不同扩展信息单元之间的语义链接，是查询语义扩展路由的基础，其中：1)key 对应本地索引扩展信息单元的 Hash 键值；2)Local EIU 亦即本地键值对应扩展信息单元名称；3)Local MST 对应本地扩展信息单元所属 MST 标识；4)Link EIU 对应语义链接信息单元名称；5)Link MST 对应语义链接信息单元所属 MST 标识；6)Link Type 对应语义链接类型；7)Similarity degree 对应 GSL 的词汇语义相似度；8) LOS 是语义链接的语义保持度(在第 3.4 节介绍)。

VIT 中包含了标识信息单元实例的索引信息，其中：1)key 对应标识核心信息单元实例的 Hash 键值；2)CIU Name

对应标识核心信息单元名称；3)CIU Value 对应标识核心信息单元值；4)IRC Addr. 对应包含特定标识信息单元实例的 IRC 地址。

3.3 MST 组群生成

每个 MST 都反映了某类资源的信息能力和语义内容。基于 MST 对 IRC 进行分组将有效提高查询的效率。这里首先对 MST 组群的概念进行定义。

定义 4 $MGroup(m)$ 为关于 m 的组群，$m \in S^{MST}$，$MGroup(m)=\{n \mid LMR(n,m)\neq\phi, n\in N^{IRC}\}$。

对于每一个组群 $MGroup(m)$ 都包含一个组长节点(Group Leader)，表示为 $GL(m)$，$GL(m)\in MGroup(m)$。

定理 1 对于 $\forall n_i \in N^{IEN}$，如果 $\exists(k,\_,m,\_,\_)\in S^{IIT}(n_i)$，有 $LIN(n_i,k,m)=MGroup(m)$，其中 $S^{IIT}(n_i)$ 表示节点 n_i 的 IIT 表记录集。

其中"_"符号表示其对应项可以取任意值，后面的"_"符号都是此含义。

证明：对于 $\forall n\in LIN(n_i,k,m)$，根据 $LIN(n_i,k,m)$ 的定义可知，必然 $\exists(k,\_,m,n,\_)\in S^{IIT}(n_i)$，因为 $(k,\_,m,n,\_)\in S^{IIT}(n_i)$，根据 $S^{IIT}(n_i)$ 定义可知 $LMR(n,m)\neq\phi, n\in N^{IRC}$，所以有 $n\in MGroup(m)$，则 $LIN(n_i,k,m)\subseteq MGroup(m)$。

对于 $\forall n\in MGroup(m)$，根据 $MGroup(m)$ 定义，可知 $LMR(n,m)\neq\phi$，因为 $\exists(k,\_,m,\_,\_)\in S^{IIT}(n_i)$，则 $\exists c\in MIE(m)$，$s.t.\ k=hash(c)$，$k\in KeyRange(n_i)$，又因 $LMR(n,m)\neq\phi$，根据 $S^{IIT}(n_i)$ 定义可知 $(k,\_,m,n,\_)\in S^{IIT}(n_i)$，因而 $n\in LIN(n_i,k,m)$，则 $MGroup(m)\subseteq LIN(n_i,k,m)$。

因为 $LIN(n_i,k,m)\subseteq MGroup(m)$ 且 $MGroup(m)\subseteq LIN(n_i,k,m)$，则 $LIN(n_i,k,m)=MGroup(m)$。

证毕。

根据定理 1 的结论，在 M-Chord 中利用 IEN 中 IIT 表的索引信息来构建 MST 组群。MST 组群构建过程如下：

1)任意 $n\in N^{IEN}$，会周期性地检查本地 IIT 表。对于任意 $(k,\_,m,\_,\_)\in S^{IIT}(n_i)$，如果 $|LIN(n_i,k,m)|\geqslant\lambda$($\lambda$ 预先给定阈值)，且 GIT 表中不存在记录 $(m,\_)$ 则进入 2)，触发 $MGroup(m)$ 的构建过程，否则退出组群构建过程；

2)在 $LIN(n_i,k,m)$ 中选择 $GL(m)$；

3)将 $LINS(n_i,k,m)$ 信息发送给 $GL(m)$，$GL(m)$ 返回确认消息，并依据 $LINS(n_i,k,m)$ 中的信息生成组播树，具体方法下一节进行介绍；

4)n_i 收到确认消息后，将 $(m,GL(m))$ 加入到 GIT 表中，并将 $(m,GL(m))$ 消息传播给 N^{IEN} 中所有节点。

在 MST 组群构建过程第 2)步中 $GL(m)$ 的确立主要是依赖 $LINS(n_i,k,m)$ 中 State(n) 信息，State(n) 中包含 IRC 节点 n 的节点状态信息。为了计算节点能力，$NC(n)$ 需要包括的状态参数有：1)日均在线时间(Uptime Per Day，UPD)；2)最大处理能力(Max. Processing Power，MPP)；3)日均处理能力可用率(Avg. ratio of Available Processing power per day，AAP%)；4)最大网络带宽(Max. Network Bandwidth，MNB)；5)日均网络带宽可用率(Avg. ratio of Available network Bandwidth per day，AAB%)；6)可用存储空间(Available Storage Space，ASS)。针对这些参数值，提供节点能力 $NC(n)$ 计算公式如下：

$$NC(n)=\alpha_1\frac{UPD(n)}{\mathrm{Max}DT}+\alpha_2\frac{MPP(n)\cdot AAP\%(n)}{\mathrm{Max}PP}+\alpha_3\frac{MNB(n)\cdot AAB\%(n)}{\mathrm{Max}NB}+\alpha_4\frac{ASS(n)}{\mathrm{Max}SS}$$

其中，$\alpha_1,\alpha_2,\alpha_3,\alpha_4$ 为可调节参数，$\alpha_1+\alpha_2+\alpha_3+\alpha_4=1$ 且 $\alpha_1\geqslant\alpha_2\geqslant\alpha_3\geqslant\alpha_4\geqslant 0$，后者反映了各个因素在表现节点能力上起到的作用依次递减。MaxDT，MaxPP，MaxNB，MaxSS 是系统预定义常量，表示所有网络节点中最大在线时间、最大处理能力、最大网路带宽、最大存储空间。

在 MST 组群构建过程中，$GL(m)$ 的初始选择过程由 IEN 完成，确立后由新任 $GL(m)$ 完成。IEN 通过 State(n) 中提供的信息分别计算每一个 $n\in LIN(n_i,k,m)$ 的 $NC(n)$，并选择其中 NC 值最大的节点为 $GL(m)$。

$GL(m)$ 会周期性地根据收集的组成员状态信息生成组播树，计算树延迟 D_{new}，并与当前树延迟 D_{cur} 进行比较。若 $D_{new}-D_{cur}>T$(T 为预定阈值)，则用新的组播树结构构建成员拓扑。

3.4 语义扩展路由

IEN 节点可以基于语义链接将资源发现请求路由到语义相关的 MST 组群或节点。下面首先对语义扩展路由的相关概念进行定义。

定义 5 语义链接(Semantic Link，SL)是信息单元之间的显式和隐式语义关系定义。包括：

1)扩展语义链接(Extensional Semantic Link，ESL)。对于不同的 $m_1,m_2\in S^{MST}$，假定 m_2 是在 m_1 上的扩展，则 m_2 包含 m_1 中所有信息单元定义。如果存在信息单元 $c_1\in MIE(m_1)$，$c_2\in MIE(m_2)$，且存在 $H^{IU}(c_1,c_2)$，则称 c_1,c_2 之间存在扩展语义链接，可表示为 $ESL(c_1,c_2)$ 或 $ESL(c_2,c_1)$。

2)等价语义链接(eQuivalence Semantic Link，QSL)。对于不同的 $m_1,m_2\in S^{MST}$，若存在如定义 2 中的 $MR^{m1\to m2}$ 定义，且存在 $c_1\in MIE(m_1)$，$c_2\in MIE(m_2)$，$c_1\succ\prec c_2$，则称 c_1,c_2 之间存在等价语义链接，可表示为 $QSL(c_1,c_2)$ 或 $QSL(c_2,c_1)$。

3)包含语义链接(Subsumption Semantic Link，SSL)。对于不同的 $m_1,m_2\in S^{MST}$，若存在如定义 2 中的 $MR^{m1\to m2}$ 定义，且存在 $c_1\in MIE(m_1)$，$c_2\in MIE(m_2)$，$c_1\succ c_2$，则称 c_1,c_2 之间存在包含语义链接，可表示为 $SSL(c_1,c_2)$ 或 $SSL(c_2,c_1)$。

4)重叠语义链接(Overlap Semantic Link，OSL)。对于不同的 $m_1,m_2\in S^{MST}$，若存在如定义 2 中的 $MR^{m1\to m2}$ 定义，且存在 $c_1\in MIE(m_1)$，$c_2\in MIE(m_2)$，$c_1\sim c_2$，则称 c_1,c_2 之间存在重叠语义链接，可表示为 $OSL(c_1,c_2)$ 或 $OSL(c_2,c_1)$。

5)词汇语义链接(Glossary Semantic Link，GSL)。即信息单元 c_1,c_2 命名词汇语义相似度 NameSim(c_1,c_2) 大于特定阈值 λ_{min}，则称 c_1,c_2 之间存在词汇语义链接，可表示为 $GSL(c_1,c_2)$ 或 $GSL(c_2,c_1)$。

其中 1)、2)、3)、4)是显式语义链接(Declared Semantic Link，DSL)，是在元数据知识库中明确定义了的；5)是隐式语义链接(Undeclared Semantic Link，USL)，没有在元数据知识库中明确定义，需要通过计算获得。

定义 6 对于 $\forall c_1\in MIE(m_1)$，若存在 $SL_1(c_1,c_2)$，$SL_2(c_2,c_3)$，$\cdots$，$SL_k(c_k,c_{k+1})$，$SL_1,\cdots,SL_k\in\{ESL,QSL,SSL,OSL,GSL\}$，$c_2\in MIE(m_2)$，$\cdots$，$c_{k+1}\in MIE(m_{k+1})$，且 $m_1,\cdots,m_{k+1}$ 两两不同，则称 c_1 与 c_{k+1} 之间存在语义可达路径(Reachable Semantic Path)，表示为 $RSP(c_1,c_{k+1})=\{SL_1(c_1,c_2),SL_2(c_2,c_3),\cdots,SL_k(c_k,c_{k+1})\}$。$|RSP(c_1,c_{k+1})|$ 表示

$RSP(c_1,c_{k+1})$语义可达路径长度，$|RSP(c_1,c_{k+1})|=k$。

对于不同语义链接赋予一个链接语义保持度来反映使用相应语义链接的语义损耗情况。值越小，损耗越大。$LOS(SL)\in[0,1]$。初始状态下，链接语义保持度 $LOS(SL(c_i,c_j))=LOS_{init}(SL(c_i,c_j))$，$LOS_{init}(SL(c_i,c_j))$为：

$$LOS_{init}(SL(c_i,c_j))=\begin{cases}\alpha_{ESL}, & SL=ESL\\ \alpha_{QSL}, & SL=QSL\\ \alpha_{SSL}, & SL=SSL\\ \alpha_{OSL}, & SL=OSL\\ \alpha_{GSL}\cdot \mathrm{NameSim}(c_i,c_j) & SL=GSL\end{cases}$$

其中，$\alpha_{ESL},\alpha_{QSL},\alpha_{SSL},\alpha_{OSL},\alpha_{GSL}$是语义保持度初始系数，$\alpha_{QSL}>\alpha_{ESL}>\alpha_{SSL}>\alpha_{OSL}>\alpha_{GSL}$。

链接语义保持度 $LOS(SL(c_i,c_j))$会随着查询过程中 $SL(c_i,c_j)$的使用反馈不断更新，IEN 节点会周期性地根据其管理的语义链接 $SL(c_i,c_j)$的使用反馈信息计算新的链接语义保持度 $LOS(SL(c_i,c_j))$，计算方法如下：

$$LOS(SL(c_i,c_j))=LOS_{init}(SL(c_i,c_j))\cdot\frac{Helpcnt(SL(c_i,c_j))}{Usedcnt(SL(c_i,c_j))}$$

其中，$Usedcnt(SL(c_i,c_j))$表示使用语义链接 $SL(c_i,c_j)$的次数，而 $Helpcnt(SL(c_i,c_j))$表示使用 $SL(c_i,c_j)$获得查询结果的次数。

$RSP(c_1,c_{k+1})$的语义保持度 $LOS(RSP(c_1,c_{k+1}))=\prod_{i=1}^{k}LOS(SL_i(c_i,c_j))$。

定义 7 对于涉及扩展信息单元 c_1 的资源发现请求 $Q(c_1)$，若存在 $RSP(c_1,c_{k+1})=\{SL_1(c_1,c_2),SL_2(c_2,c_3),\cdots,SL_k(c_k,c_{k+1})\}$，$c_1\in MIE(m_1),c_2\in MIE(m_2),\cdots,c_{k+1}\in MIE(m_{k+1})$且$|RSP(c_1,c_{k+1})|\leqslant L$，$LOS(RSP(c_1,c_{k+1}))\geqslant U_{min}$，则将 $Q(c_1)$路由到 $GL(m_1),GL(m_2),\cdots,GL(m_{k+1})$进行进一步匹配，称这一过程为基于 c_1 的语义扩展路由。

其中，U_{min}为最小语义保持度阈值，L 为最大语义可达路径长度约束，这些参数都可以在查询前进行设置，来控制语义扩展路由的范围。而且 IEN 节点周期性地更新不同语义链接的语义保持度，可以有效地减少不必要的语义扩展路由。

4 语义检索方法

本节首先对语义查询消息结构设计如下：

定义 8 语义查询消息结构 $QM=query(ID,q,InitialNode,State,Key,MST,TT,L,U_{max},\lambda_{min})$，其中 ID 为查询标识，唯一确定一个查询；$q=(m,qs,qr,qc)$，其中 m 表示查询定义参考的 MST，当 $m=$NULL 时表示查询没有参照任何 MST，是定义的一个新的需求本体；qs 表示查询中涉及的信息单元集合；qr 表示查询中涉及的信息单元关系集合；qs,qr 的结构定义与元数据抽象模型中 S^{IU},R^{IU} 的结构定义相同；qc 定义了查询中信息单元实例必须满足的约束条件；$InitialNode$ 是发起查询的 IRC；$State$ 表示查询处理过程所处状态；Key 是查询路由的当前索引键值；MST 是查询当前组播所属的 MST 组群 ID；TT 是语义路由扩展轨迹，$TT=\{(m_1,c_1,m_2,c_2,sl_1),(m_2,c_2,m_3,c_3,sl_2),\cdots,(m_k,c_k,m_{k+1},c_{k+1},sl_k)\}$，$m_1,\cdots,m_{k+1}\in S^{MST}$，$c_1\in MIE(m_1),c_2\in MIE(m_2),\cdots,c_{k+1}\in MIE(m_{k+1})$，$sl_1,sl_2,\cdots,sl_k\in\{ESL,QSL,SSL,OSL,GSL\}$；$L$ 为最大语义可达路径长度约束；U_{max}为最大语义保持度阈值；λ_{min}表示资源发现请求与 MST 的初级匹配度的最低阈值。

M-Chord 中语义检索方法的过程：1)对于语义查询 Q，首先判断查询条件中是否存在标识信息单元。存在，则直接利用 Chord 协议在 IEN 节点中发现资源位置，返回结果。2)如果语义查询 Q 中不涉及标识信息单元，存在可索引的语义信息单元，则基于这些语义信息单元词汇定义，采用 Chord 协议在 IEN 中找到包含这些语义信息单元的 MST 组群。同时基于语义信息单元之间存在的语义链接，进行语义扩展路由，将查询发送到语义相关的 MST 组群。3)通过组群组长节点判断是否向组群组播请求。当查询与相应 MST 初级匹配度[11]满足特定阈值时，向组群组播查询，采用基于语义相似度的资源匹配方法[11]，定位满足查询需求的资源，返回给用户。4)若不存在可索引的信息单元，则采用全局范围内的泛洪策略，搜索范围由 TTL 限制。

5 实验分析

根据 M-Chord 体系环境设计，对 Chord 源码进行必要修改，生成相应仿真试验平台 RA_Chord。实验环境中用 10 台奔腾 IV、内存 512M 的 PC 机模拟 1000 个 IRC 节点。每台 PC 机模拟 100 个 IRC 节点。每个节点都维护一个消息队列，用于存储其它节点发送来的各种消息。各节点的本地时间推进采用随机步长，各节点通过周期性地以一定的间隔在全局事件表中插入自己的消息处理事件来处理消息队列中的消息，而这一间隔是一个服从一定范围内均匀分布的随机数。本机节点间的消息传递采用一定范围内的随机延迟进行模拟，保持与不同机器间消息传递延迟比例相当。在仿真环境中，针对教学、科研领域设计 200 个 MST，MST 中包含语义信息单元 10～20 个，并模拟 4000 个资源，分别参考 200 个 MST 进行描述。相对均匀地将资源随机分配到各个 IRC 节点上，并建立不同 MST 组群，选取组长节点 GL，并建立 MST 组播树(树的最大度约束 $d_m^{max}(v)=4$)。在 1000 个 IRC 中选择 32 个节点作为 IEN，根据 Chord 协议模拟 IEN 节点之间的一维环状拓扑结构。基于信息单元键值在 IEN 上建立索引，并预先定义一组 MST 之间的映射规则，依据这些规则和不同扩展信息单元之间的命名相似度，在每个 IEN 上预先建立语义链接表，语义链接 SL 的语义保持度 $LOS(SL)$设置在 0.6～1.0 之间。同时，为了对比，由于现有的基于语义模式的 P2P 系统主要采用一种无结构的拓扑网络，因而试验采用一种类 Gnutella 松散无结构的对等网络，采用和前面 M-Chord 中一样的网络拓扑结构，忽略其中不同的节点角色和连接类型。而资源发现的路由策略采用 k 随机步(Random Walker)方案来进行模拟。随机步方案就是每个随机步随机地选择当前节点的一个邻节点转发查询请求，直至达到相应的 TTL 值或当前节点对所有邻节点都转发过请求为止。

如图 4 所示，RA_Chord 随着节点规模的扩大，对查全率(Recall)的影响不大，而且在节点规模扩大的情况下性能明显优于 Random_walk 方法。同时可以看出，资源密度对 RA_Chord 中查全率的影响不大，而 Random_walk 方法随着资源密度的减小，查全率明显降低。

如图 5 所示，RA_Chord 相比 Random_walk 方法，在较少网络消息量的基础上，可以更快地定位所需的信息资源。总体来说，RA_Chord 相对 Random_walk 在具有较小网络开销

(下转第 69 页)

情况下的视频数据传输服务。

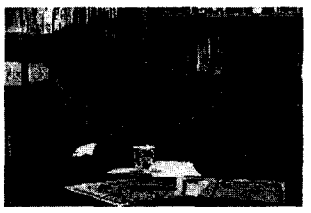

图5 合成1个信道 PSNR=18.73

图6 合成2个信道 PSNR=20.08

图7 合成3个信道 PSNR=23.09

图8 合成全部信道 PSNR=39.09

参 考 文 献

[1] Radha H M, van der Schaar M, Chen Yingwei. The MPEG-4 fine-grained scalable video coding method for multimedia streaming over IP[J]. IEEE Transactions on Multimedia, 2001, 3(1): 53-68

[2] Tham Chen-Khong, Jiang Yuming, Gan Yung-Sze. Layered coding for a scalable video delivery system[C]// Proceedings of IEEE/EURASIP Packet Video 2003 (PV 2003). 2003: 28-29

[3] Reibman A R, Jafarkhani H, Wang Yao, et al. Multiple description coding for video using motion compensated prediction[C]// Image Processing, 1999. ICIP 99. Proceedings. 1999 International Conference. Volume 3, Oct. 1999: 837-841

[4] Apostolopoulos J G. Reliable video communication over Lossy packet networks using multiple state encoding and path diversity [M]. Visual Communications and Image Process. San Jose: [s. n.], 2001: 392-409

[5] Goyal V K. Multiple description coding: compression meets the network[J]. IEEE Signal Processing Magazine, 2001, 18(5): 74-93

[6] Albanese A, Blömer J, Edmonds J, et al. Priority encoding transmission[J]. IEEE Trans. Information Theory, 1996, 42: 1737-1744

[7] Wang Huisheng, Ortega A. Robust video communication by combining scalability and multiple description coding techniques [A]// Proceedings of the SPIE[C]. 111-124

(上接第 64 页)

的基础上，具有更高、更快的查询效率。

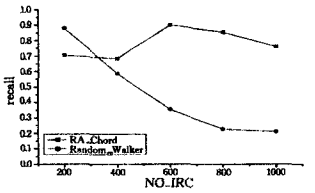

图4 不同节点规模下的查全率比较

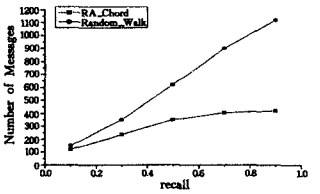

图5 查询消息量比较

如图6所示，针对10个查询进行统计，每个查询中随机选取4～8个语义信息单元。取最小语义保持度阈值 $U_{min}=0.30$，分别给出了取 $L=0,2,5,10$ 时不同时间内的平均查全率(recall)。从实验结果可以看出，当 $L=0$，即不进行语义扩展路由时，查全率受到了很大的影响，总体查全率较低。随着 L 的增长，查全率会随之增高。L 增长到一定范围，对查全率影响将不会很大，这主要是由于可供语义扩展路由的链接数量和语义损耗的限制。

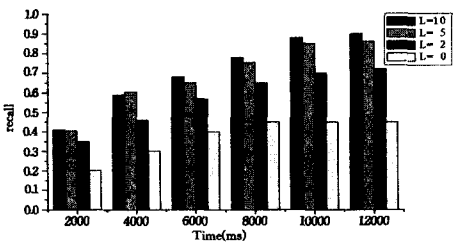

图6 语义扩展路由对查询性能的影响

结束语 本文提出一种支持语义的混合 P2P 网络模型 M-Chord。采用基于 MST 的语义描述模型，通过扩展 Chord 协议对资源的语义描述信息单元进行索引，并通过 MST 组群的动态生成，形成语义覆盖网，进行索引。通过语义信息单元之间的语义链接，提出语义扩展路由的概念。在此基础上设计语义检索算法，为构建基于本体的对等网语义检索系统提供了理论支持。通过实验对比分析，证明 M-Chord 可扩展性强，应用语义检索有效提高了系统的检索性能，查全率和查准率得到明显改善。

参 考 文 献

[1] Purushothaman P, Navada M, et al. Power - proxying on the NIC: A Case Study with the Gnutella File-sharing Protocol[C] // Local Computer Networks, Proceedings 2006. IEEE, Nov. 2006

[2] Skogh H, Haeggstrom J, et al. Fast Freenet: Improving Freenet Performance by Preferential Partition Routing and File Mesh Propagation[C]// Cluster Computing and the Grid Workshops. 2006

[3] SWAP Research Community. SWAP EU IST-2001-34103 Final Report[R]. http://swap.semanticweb.org/public/public/ Publications/finalReport.pdf, 2007-06-06

[4] Castano S, Ferrara A, Montanell I S, et al. Helios : a general framework for ontology-based knowledge sharing and evolution in P2P systems[C]// Proc. of the 14th International Workshop on Database and Expert Systems Applications. 2003

[5] 凌波，陆志国，黄维雄，等. PeerIS：基于 Peer-to-Peer 的信息检索系统[J]. 软件学报，2004，15(19)：1375-1384

[6] 金海，陈汉华，宁小敏，等. SemreX 系统中一种基于语义相似度的 Peer-to-Peer 拓扑及路由算法[C]//CNCC2005. 北京：清华大学出版社，2005

[7] 陈汉华，金海，宁小敏，等. SemreX：一种基于语义相关度的 P2P 覆盖网络[J]. 软件学报，2006，17(5)：1170-1181

[8] 宋建涛，沙朝锋，杨智应，等. 语义对等网构造及搜索机制研究[J]. 计算机研究与发展，2004，41(4)：645-652

[9] 于婧，汪斌强. SSON：一种基于结构化 P2P 网络路由的语义覆盖网络结构[J]. 计算机科学，2007，34(6)

[10] Yu Yun-shuai, Miao Yu-ben, Shieh Ce-kuen. Improving the Lookup Performance of Chord Network by Hashing Landmark Clusters[C]// ICON '06. Sept. 2006

[11] 刘震，邓苏，罗雪山，等. 基于多本体语义相似度计算的对等网资源动态匹配算法研究[J]. 计算机科学，2006，33(3)