

蚁群算法在 P2P 搜索中的应用研究

郭玉龙, 吴晓军

(陕西师范大学 计算机科学学院, 陕西 西安 710062)

摘要:为了提高 P2P 网络中资源的搜索效率, 蚁群算法被广泛应用于 P2P 网络的搜索中。但 P2P 网络中搜索的是大量不同的文件, 不同于蚂蚁寻找的单一食物, 目前的研究较少考虑到这方面的不同。基于此在应用基本蚁群算法的基础上, 引入合成信息素的概念, 提出了合成信息素的提取策略, 并编程进行了模拟仿真。实验结果表明相比洪泛和基本蚁群算法, 可提高查找的结果数并降低网络中产生的消息包总数。因此在蚁群算法中引入合成信息素可有效提高 P2P 网络的查找效率, 进而提高网络的性能。

关键词:对等网; 搜索; 蚁群算法; 合成信息素

中图分类号: TP393

文献标识码: A

文章编号: 1673-629X(2011)10-0055-04

Research on Search Based on Ant Colony Algorithm in P2P Networks

GUO Yu-long, WU Xiao-jun

(School of Computer Science, Shaanxi Normal University, Xi'an 710062, China)

Abstract: In order to improve the resources search efficiency in peer-to-peer networks, ant colony algorithm has been applied to peer-to-peer networks widely. But there are lots of files to be searched, which different from the signal resource in ant colony. At present this difference is less considered. So import the conception of generated pheromone based on ant colony algorithm, and the produce strategy of generated pheromone. Simulate the algorithm resented and the experiment result shows that ant algorithm with generated pheromone can improve the search result account but decrease the search packages account. So import generated pheromone in basic ant algorithm can improve the search efficiency, and the network performance at the same time.

Key words: peer-to-peer network; search; ant colony algorithm; generated pheromone

0 引言

P2P (Peer-to-Peer) 网络也即对等网络^[1], 是不同于传统的 C/S (客户机/服务器) 模式的一种新的网络应用模型, 在 P2P 网络中不再有客户机、服务器的区别, 所有的节点的地位和功能都是平等的, 节点既是资源的使用者又是资源的提供者, 因此不存在传统 C/S 网络应用模型中的服务器瓶颈问题。按照节点组织方式的不同, 可将 P2P 网络总体上分为三种模型: 混合模型、结构化模型、非结构化模型^[2]。在混合模型中节点之间直接传输数据, 但搜索在索引服务器中进行, 仍然不能摆脱对服务器的依赖; 在结构化模型中通过 DHT (Distributed Hash Table) 将资源映射到不同的节点上, 摆脱了对服务器的过度依赖, 但由于网络的动态

性, 维护 DHT 的开销较大, 同时由于资源标识是经过哈希函数计算后得到的, 失去了原来的意义, 因此对模糊检索的支持程度比较低^[3]; 在非结构化模型中节点没有明确的组织结构, 搜索一般基于洪泛机制来实现, 由于洪泛的盲目性, 在网络中会产生很多查询包, 对网络带宽的消耗比较大^[4]。非结构化模型中节点随机地互相连接在一起, 节点完全自组织, 很多模型都是基于此衍生出来的, 无结构化 P2P 网络理论受到研究人员广泛关注。

1 相关研究

搜索是 P2P 网络必须提供的基本功能, 也是 P2P 网络的难题, 目前已有许多基于 P2P 搜索的研究。在无结构 P2P 网络中, 洪泛是最基本也是最简单的搜索方法, 查询发起节点将查询请求发送给自己的邻居, 收到查询请求的节点如果没有在本地找到请求的资源则也转发给自己的邻居, 查询过程递归地进行下去直到找到资源或者请求消息包达到最大跳数限制查询失败。洪泛会产生大量的查询包, 占用大量的网络带宽, 降低了网络的性能^[5], 因此需要设法减少查询包数量,

收稿日期: 2011-02-25; 修回日期: 2011-05-06

基金项目: 陕西省工业攻关计划 (2009K09-21); 中央高校基本科研业务费专项资金资助项目 (GK201002005)

作者简介: 郭玉龙 (1984-), 男, 河南人, 硕士研究生, 研究方向为计算机应用; 吴晓军, 硕士生导师, 副教授, 研究方向为计算机应用、模式识别。

主要是控制消息的跳数和每次转发的邻居数。

随机走法就是每次只向一个邻居转发查询包,每个查询包相当于一个行走者。扩展环法则是逐渐增大 TTL,类似广度优先搜索的过程^[6]。基于洪泛的搜索是盲目的,产生的查询包消耗的网络带宽比较大,效率比较低,因此研究者提出了启发式搜索算法^[7]。在启发式搜索过程中,查询消息转发到最可能存在请求资源的节点上,启发信息的建立和维护成为启发式算法的关键。

研究发现 P2P 网络和群体生活的蚂蚁^[8]具有很多相似之处,蚁群算法被应用到 P2P 网络中以提高搜索效率^[9]。但 P2P 网络中的搜索和蚂蚁寻找食物并不完全相同,在蚁群中蚂蚁寻找的资源只有一种即食物,在 P2P 网络中搜索的资源却是不同的,寻找的是大量不同的文件,目前的研究较少考虑到这方面的区别,本论文的研究基于此展开。

2 蚁群算法及合成信息素在 P2P 搜索中的应用

2.1 蚁群算法及其在 P2P 搜索中的应用

蚁群算法^[10]最初是由意大利学者 M. Dorigo 等人通过观察自然界中蚂蚁的觅食行为,于 20 世纪 90 年代初提出的一种新的群体智能优化算法^[11]。蚂蚁在觅食的时候会在其所经过的路径上释放一种被称为信息素(pheromone)的物质,并且能够感知信息素的存在以及强度。后来的蚂蚁会根据信息素的强度来选择路径,这样,当某条路径上走过的蚂蚁越多,留下的信息素也就越多,蚂蚁再次选择该条路径的可能性也就越大;信息素随着时间的推移会逐渐挥发从而强度逐渐降低,对于没有或者很少有蚂蚁经过的路径,信息素的强度会越来越低,后来的蚂蚁选择这条路径的可能性也就越小。信息素实际上构成了一种正反馈机制,通过这种机制,蚂蚁最终可以发现到达食物的最优路径。

在非结构化 P2P 网络中资源分散在网络中的每一个节点上,搜索通过洪泛机制,查询在节点之间不停地进行转发,最后找到目标资源。搜索可以看作是蚂蚁寻找食物的过程,查询发起节点发出的查询请求包可以看成是一个蚂蚁,搜索的过程就是找到一条到达目标节点的最优路径的过程^[12]。为了减少节点转发查询包的盲目性,节点需要记录其邻居节点的情况,为搜索提供指导信息,这些信息相当于蚂蚁留下的信息素。

对于节点 i 引入集合 N_i 记录其邻居节点,集合 F_i 记录查找过的文件的信息素, F_i 中的元素 f_{ik} 记录查找文件 k 时邻居节点 j 的响应情况,相当于信息素,集合 tabu_i 记录已经转发过查询包的邻居节点集,节点 i 收

到查询请求包时转发给邻居 j 的概率计算公式如下:

$$p_{ij}^k = \begin{cases} \frac{f_{ik}^\alpha \eta_{ij}^\beta}{\sum_{j \in \text{tabu}_i} f_{isk}^\alpha \eta_{is}^\beta}, & \text{若 } j \notin \text{tabu}_i \\ 0, & \text{否则} \end{cases} \quad (1)$$

式中, α 为信息启发式因子,反映积累的信息素在搜索过程中的重要性; β 为启发式因子,表示能见度的相对重要性,值越大则越接近于贪心算法。

完成一次文件的搜索后,返回查找结果路径上的各个节点信息素按以下规则进行更新:

$$f_{ijk} = \begin{cases} \rho * f_{ijk}^\circ + (1 - \rho) * \Delta f_{ijk}, & \text{经过 } i, j \\ \rho * f_{ijk}^\circ, & \text{否则} \end{cases} \quad (2)$$

式中, f_{ijk}° 表示在进行本次查找前该条路径上的信息素强度; ρ 为系数,通过调整 ρ 可以调整信息素的挥发速度和更新强度, ρ 越大历史信息所起的作用越大,相应的当前信息所起的作用也就越小; Δf_{ijk} 表示节点 i 在查询文件 k 时留在该条路径上的信息素浓度,计算公式如下:

$$\Delta f_{ijk} = \begin{cases} \frac{Q}{L_k}, & \text{经过 } i, j \\ 0, & \text{否则} \end{cases} \quad (3)$$

式中 L_k 表示本次查找走过的路径总长度, Q 为常数,表示一次查找蚂蚁可释放的信息素的总量。

2.2 合成信息素及提取策略

查询进行一段时间后,信息素已经建立起来,查询速度大幅度提高,但某些文件可能仍然是第一次查找,如果按照常规方法查找,则相当于这个文件的信息素仍然没有建立起来,只能进行盲目的随机转发,查询速度将受到影响,为此引入合成信息素。合成信息素即是从查找其它文件的信息素中提取出来以表征邻居节点特性的信息素,以为没有建立起信息素的文件查询请求提供参照。合成信息素主要从响应速度方面考虑,即认为能快速返回其它文件查询结果的节点也能快速返回本次查询结果。用集合 F_{ij} 表示节点 i 到其邻居 j 的所有文件信息素,显然有 $F_{ij} \subset F_i$, 用集合 T_i 记录节点 i 的所有合成信息素, T_i 中的元素 t_{ij} 表示节点 i 到节点 j 的合成信息素, t_{ij} 的计算公式如下:

$$t_{ij} = \begin{cases} \frac{\sum_{f_{ik} \in T_j} f_{ik}}{|F_{ij}|}, & \text{若 } F_{ij} \text{ 非空} \\ 0, & \text{否则} \end{cases} \quad (4)$$

式中, $|F_{ij}|$ 表示集合 F_{ij} 中元素的个数,节点 i 到其邻居节点 j 的合成信息素主要与这个邻居节点在以往的查询过程中成功返回查询结果的平均时间有关。通常情况下性能良好的节点对每次查询请求都能以较快的速度返回。

2.3 应用合成信息素的 P2P 搜索算法

为了在 P2P 网络中应用蚁群算法,每个节点需要建立如下数据表(见表 1~表 6),以保存网络的状态信息。

表 1 邻居节点表

数据项	说明
PeerID	记录邻居节点的标识
Timestamp	更新时间戳

邻居节点表用于维护网络的拓扑结构,并可根据更新时间戳周期性地检测邻居节点是否仍然在线,以使节点了解网络的最新状态。

表 2 本地文件表

数据项	说明
FileName	文件名
Description	文件的描述信息

本地文件表记录本地存放的文件名称及描述信息,以便在查询请求到达时快速在本地完成查找操作。

表 3 文件信息素表

数据项	说明
FileName	文件名
PeerID	记录邻居节点的标识
Pheromone	信息素
Timestamp	更新时间戳

信息素表记录单个文件经过特定的邻居节点的查找信息素,当对相同的文件再次进行查找时,可根据信息素进行查询包的转发,提高查找效率。同时信息素表中记录信息素的更新时间,以便检查信息素是否已超时失效。

表 4 消息来源节点缓存表

数据项	说明
PackID	消息包标识符
PeerID	记录邻居节点的标识
Timestamp	消息接收时间戳

消息来源节点缓存表记录最近一段时间内所收到的查询消息的来源节点地址,以便收到对该消息包的响应时能确定下一跳的节点地址,使响应消息按原路返回。节点根据时间戳来判断记录是否已超时,超时的节点记录会被删除,这样即使此后收到这个消息包的响应也将其丢弃,不再进行转发,本次查找也宣告失败。

表 5 消息转发缓存表

数据项	说明
PackID	消息包标识符
PeerID	记录邻居节点的标识
Timestamp	消息转发时间戳

消息转发缓存表记录最近一段时间内收到的查询包的标识符及转发情况,当再次收到重复的查询包时将直接丢弃,以防止重复转发相同的数据包。同时超过一定时间后的记录也会被删除,即间隔较长时间的查询认为是发起的一次新的查询,仍按正常情况进行转发。

表 6 合成信息素表

数据项	说明
PeerID	邻居节点标识
Pheromone	信息素

合成信息素表为临时表,用于查询进行一段时间后对新文件的初次查询,根据以往查找其它文件时的信息素来推断当前文件的信息素。每次使用时其内容都要根据当前的情况实时生成。

通过节点上的各个数据表可以保存网络的状态,当 P2P 网络中节点 i 收到查询文件 k 的请求后的处理过程如下:

步骤 1:判断最近一段时间内是否收到相同标识符的查询包,如有转步骤 8;

步骤 2:在本地查找是否存在请求的文件 k ,若存在,转步骤 7;

步骤 3:若 $tub_{u_i} \neq N_i$ 且存在 $f_{jk} \in F_{ij}$, $f_{jk} \notin tub_{u_i}$, 将请求包转发至邻居节点 s , 使得 $f_{sk} = \min(F_{ij} - tub_{u_i})$, 转步骤 8;

步骤 4:若存在节点 $j \in N_i$ 且 $j \notin tub_{u_i}$, 根据公式 (4) 计算合成信息素 t_{ij} , 将请求包转发至节点 s , 使得 $t_{sk} = \min(T_i)$, 转步骤 8;

步骤 5:若 $N_i - tub_{u_i}$ 非空,随机选择邻居节点 $s \in (N_i - tub_{u_i})$ 进行转发;转步骤 8;

步骤 6:返回查询失败消息,转步骤 8;

步骤 7:按原路返回查询结果,并依照公式 (2) 更新新路径上节点的信息素;

步骤 8:结束。

为了防止节点重复转发查询包,对每个查询包需要加入一个标识符,节点缓存最近转发过的查询包,当相同标识符的查询包到达时直接将其丢弃。

3 实验仿真

在 VC 环境下编程模拟仿真基于洪泛机制、基本蚁群算法、应用合成信息素的蚁群算法的 P2P 搜索方法,在 400 节点的情况下查找成功次数及查询请求转发的总次数随查询重复次数的变化见表 7、表 8。

从表 7 和表 8 中可以看出,洪泛随着查询的进行,成功率没有明显变化,节点转发的查询包总数也没有明显的变化,这主要是由于洪泛转发查询包时的盲目性所决定的,历史信息对洪泛算法不产生影响;蚁群算

法随着查询的进行,成功率在逐渐上升,转发的查询包总数在逐渐降低,这与信息素的导向作用有关,随着查询的进行信息素逐渐建立起来,查询包转发的随机性降低,减少了不必要的查询包转发,效率相比洪泛有了较明显的提高;引入合成信息素后,随着查询的进行,查询成功率得到进一步的提高,查询包转发的总次数则进一步降低,这与查询进行一段时间后,合成信息素对一些文件的初次查询的导向作用有关,利用合成信息素,一些新文件在初次查找时可在一定程度上降低消息包转发的盲目性,提高查找的成功率,降低网络中消息包的总数量。

表 7 查询成功数随请求次数变化情况

重复次数	1	3	5	7	9
算法 1	52	58	44	50	56
算法 2	40	94	122	149	160
算法 3	39	128	174	209	222

表 8 查询包转发次数随请求次数变化情况

重复次数	1	3	5	7	9
算法 1	3472	3441	3554	3497	3485
算法 2	3557	3144	2989	2768	2692
算法 3	3516	2448	2145	2060	2168

注:算法 1:洪泛;算法 2:基本蚁群算法;

算法 3:引入合成信息素后的蚁群算法。

综合分析可以看出,引入合成信息素的蚁群算法可有效提高 P2P 网络查找成功的次数,降低网络中消息包的数量,对 P2P 网络搜索性能的提升效果明显。

4 结束语

分析 P2P 网络和蚁群的不同特点,根据 P2P 网络查找很多不同的文件与蚂蚁寻找的单一食物的不同,在蚁群算法的基础上引入合成信息素的概念,将蚁群

算法应用的 P2P 网络的搜索过程中。理论分析和模拟实验结果表明,提高了搜索的成功率,降低了网络中消息包的数量,提高了搜索的效率,有利于提高 P2P 网络的整体性能。

参考文献:

- [1] 吴国庆. 对等网络技术研究[J]. 计算机技术与发展, 2008, 18(7): 100-103.
- [2] 陈贵海, 李振华. 对等网络: 结构、应用与设计[M]. 北京: 清华大学出版社, 2007.
- [3] 李运娣, 冯 勇. 基于 DHT 的 P2P 搜索定位技术研究[J]. 计算机应用研究, 2006(10): 226-228.
- [4] 钱 宁, 吴国新. 无结构化 P2P 网络资源搜索机制研究综述[J]. 计算机科学, 2010, 37(4): 7-10.
- [5] 黄道颖, 陈 新, 张安琳, 等. P2P 网络 Guntella 模型中搜索消息的路由机制及改进研究[J]. 计算机工程与应用, 2003(25): 13-15.
- [6] 吴 宇, 虞淑瑶, 宋 成. 自适应 P2P 网络搜索算法[J]. 计算机工程, 2006, 32(19): 117-119.
- [7] 夏启志, 谢高岗. 无结构 P2P 网络搜索方法及其改进[J]. 计算机应用研究, 2005(9): 256-260.
- [8] 段海滨. 蚁群算法原理及其应用[M]. 北京: 科学出版社, 2006.
- [9] 苏 玉, 毛 力. 基于蚁群算法的非结构化 P2P 搜索机制的研究[J]. 计算机工程与设计, 2010, 31(5): 939-941.
- [10] Marco D, Vittorio M, Alberto C. Ant System: Optimization by a Colony of Cooperating Agents [J]. IEEE transactions on systems, man and cybernetics-part B: cybernetics, 1996, 26(1): 29-41.
- [11] 王艳玲, 李龙澍, 胡 哲. 群体智能优化算法[J]. 计算机技术与发展, 2008, 18(8): 114-117.
- [12] Elke M, Arno P, Gerti K. Using taxonomies for content-based routing with ants[J]. Computer Networks, 2007, 51(16): 4514-4528.
- [13] Haralick R M, Shangmugam K. Texture feature for image classification[J]. IEEE Transaction on Systems, 1973, 3(6): 768-780.
- [14] Manjunathi B S, Ma W Y. Texture Features for Browsing and Retrieval of Image Data[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1996: 837-842.
- [15] 贾淑华, 李星野. 基于相关关系的图像分类和图像检索[J]. 微计算机信息, 2009(5-3): 294-296.
- [16] 孙惠萍, 龚声蓉, 王朝晖, 等. 基于强化学习的相关反馈图像检索算法[J]. 计算机工程与应用, 2008, 44(34): 175-178.
- [17] 钱秋银, 张正兰. 一种基于多分类 SVM 的相关反馈图像检索方法[J]. 计算机技术与发展, 2009, 19(8): 65-68.

(上接第 54 页)

- [2] 韦 娜, 耿国华, 周明全. 一种新的文物图像检索方法[J]. 计算机应用, 2005, 25(8): 1789-1791.
- [3] 王 松, 蒋苏蓉, 冯 刚. 基于纹理特征的一种图像检索方法的实现[J]. 安阳师范学院学报, 2002(2): 6-8.
- [4] 冯建辉, 杨玉静. 基于灰度共生矩阵提取纹理特征图像的研究[J]. 北京测绘, 2007(3): 19-22.
- [5] Anne H, Solberg S, Jain A K. Texture Fusion and Feature Selection Applied to SAR Imagery[J]. IEEE Transaction on Geoscience and Remote Sensing, 1997, 35(2): 475-478.
- [6] 黄 晶, 杨 杰. 图像纹理特征的分析方法研究[D]. 武汉: 武汉理工大学信息工程学院, 2003.
- [7] 沈庭芝, 方子文. 数字图像处理及模式识别[M]. 北京: 北京理工大学出版社, 1998.