

一种基于蚁群算法的非结构化 P2P 网络搜索算法

蓝慧琴¹, 钟 诚¹, 李 智^{1,2}

(1. 广西大学 计算机与电子信息学院, 广西南宁 530004;

2. 广西科技信息网络中心, 广西南宁 530012)

摘 要:在类似 Gnutella 的分散的非结构化 P2P 网络中, 如何降低消息开销, 提高搜索效率, 是解决其扩展性问题的关键。引入蚁群算法的思想, 提出一个非结构化 P2P 网络搜索算法。此算法利用蚂蚁留下信息素的正反馈机制, 有效地指导搜索的方向, 将查询消息包尽量发往目标可能存在的区域, 从而减少冗余消息包的产生, 得到更好的搜索输出。

关键词:P2P 网络; 搜索算法; 蚁群算法

中图分类号:TP393

文献标识码:A

文章编号:1673-629X(2006)10-0026-03

A Search Algorithm for Unstructured Peer-to-Peer Networks Based on Ant Colony Optimization

LAN Hui-qin¹, ZHONG Cheng¹, LI Zhi^{1,2}

(1. School of Computer and Electronics and Information, Guangxi University, Nanning 530004, China;

2. Guangxi Science and Technology Information Network Center, Nanning 530012, China)

Abstract: For the scalability problem of the Gnutella-like decentralized and unstructured peer-to-peer networks, how to reduce their message packets and improve searching efficiency is very important. Based on Ant Colony Optimization principle, this paper proposes a new search algorithm for unstructured peer-to-peer networks by using the fact that the pheromone trail-laying-and-following behavior observed from natural ants applied to query routing in peer-to-peer networks and the peers will send the query messages to the areas where there could be the requested objects using the pheromone. The algorithm can reduce the produced message packets in the network, and gain the better searching results.

Key words: peer-to-peer network; searching algorithm; ant colony optimization

0 引言

P2P(peer-to-peer, 即对等)技术构建的网络, 是一种非中心化网络结构^[1]。与传统的 C/S(客户机/服务器)网络结构不同, P2P 网络结构下的各个节点既是客户机又是服务器, 它们在网络中的地位和功能是同等的, 可以任意地加入和离开网络。用户不需要通过特定的服务器, 就可跟网络中的其他用户直接进行资源共享和信息交互。P2P 网络具有集中式服务网络所缺乏的优势: 可扩展性强、容错性好、成本低、充分利用分布资源。这些特点使得 P2P 架构在文件共享、分布式存储、搜索引擎、分布式计算、传感器网络、协作软件中有广泛的应用前景^[2]。

在非结构化 P2P 网络中, 由于没有中心服务器的控制, 节点要通过彼此联系来共享资源, 而且节点可以任意

加入和离开网络。因此, 如何在这样的大规模动态的分布式系统中快速搜索出用户需要的资源, 是 P2P 网络的关键问题之一。

1 相关研究进展分析

P2P 文件共享系统(如 Gnutella)广泛采用宽度优先搜索技术。节点通过洪泛(flooding)方式, 将查询消息传播给所有的邻居节点, 实现查询结果的高命中率。但是呈指数级增长的消息包会浪费过多的网络资源和处理资源, 低带宽的节点也很容易成为瓶颈。另一方面, 随机走步(random walk)搜索算法^[3]可以避免产生大量的消息包, 它随机发起 k 个同样的查询, 每个收到查询的节点随机挑选自己的一个邻居来作为下一跳, 这样它的查询消息的数量就是恒定的, 与网络拓扑无关; 只是它可能要花更长的时间来找到用户要求的文件, 或者是无法找到网络中实际存在的文件。如何探索随机走步和洪泛搜索之间的不确定的性能空间, 来获得系统的性能提升, 是 P2P 网络搜索的关键问题。也就是说, 文中的目标是设计一个鲁棒性好的 P2P 网络搜索算法, 既能达到高命中率, 又能维持较低

收稿日期: 2006-02-15

基金项目: 广西科学基金(桂科基 0575014); 广西科技信息网络中心

作者简介: 蓝慧琴(1980-), 女(壮族), 广西忻城人, 硕士研究生, 研究方向为网络与并行计算; 钟 诚, 博士, 教授, 研究方向为网络与并行计算、网络信息安全。

的消息开销。

文献[4]提出了一个基于模拟退火的混合搜索算法,它混合应用了洪泛和随机走步两种算法,以此来解决扩展性问题和决定随机步(walker)的最佳数量;它根据节点的链接状态和搜索时间,动态地决定随机步的数量,使得决定查询消息的最佳数量成为可能;在开始的时候,它洪泛局部搜索空间,然后采用随机走步方法,控制搜索对网络全局空间的影响。文献[5]提出了基于人工免疫原理的搜索算法,假设查询消息包为抗体,搜索目标所在的节点为抗原,搜索目标的过程就是抗体追捕抗原的过程,根据消息包和所访问节点内容之间的亲和力度量(即相关性比较),对消息包进行突变和增生,来实现高效的搜索。

文献[6]设计了基于蚁丘的 P2P 系统,它是一个由互相连接的蚁巢组成的网络,每个蚁巢是一个可以计算和提供资源的对等节点;蚁巢通过产生一只或多只蚂蚁(即自治代理, autonomous agents)来处理用户请求,它们在蚁巢网络中移动以满足查询;蚂蚁借助修改它们的环境(例如修改存储在蚁巢中的信息)来互相间接地通信。文献[7]根据蚁群算法的思想,设计了一个分布式搜索引擎,其中每个节点负责管理一个与某类概念相关的文档知识库,并共享库里的内容给其他节点。当某个节点需要查询文档时,它就提交一个或多个关键字到网络中,然后搜索过程就像蚂蚁寻觅粮食一样在网络中扩散。此种搜索方法的缺点是关键字集合必须是概念集合的子集,否则无法查询;并且所有节点的时钟必须是同步的。

蚁群算法^[8]是一种新颖的进化类系统优化方法。文中将蚁群算法的思想引入非结构化 P2P 网络的搜索中,提出一种基于蚁群算法的 P2P 网络搜索算法。此算法根据蚂蚁觅食行为的特性,通过蚂蚁释放信息素的正反馈机制来指导搜索前进的方向,从而尽快地得到更好的搜索输出。

2 基于蚁群算法的非结构化 P2P 网络搜索算法

2.1 蚁群算法

蚂蚁行动时会在经过的地方留下一一种挥发性的分泌物——信息素(pheromone)。随着时间流逝,信息素逐渐挥发,降低浓度。觅食开始时,蚂蚁随机挑选路径行走,经过一段时间后,如果从源头到目的地有多条路径,那么长路径上留下的信息素浓度肯定低于短路径上留下的信息素浓度。由于蚂蚁选择信息素浓度高的路径行走的概率远大于信息素浓度低的路径,所以随着时间的流逝就形成一个正向反馈,即最优路径上的信息素浓度越来越大,而其他路径上的信息素浓度逐渐减少直至消失,最终整个蚁群找到觅食的最佳路径。也就是说,生物世界中的蚂蚁有能力在没有任何可见提示下找出从蚁穴到食物源的最短路径,并且能随环境的变化而变化地搜索新的路径,产生新的选择。

受自然界中蚁群行为的启发,意大利学者 M. Dorigo

于 1991 年首次提出了一种基于蚂蚁种群的新型优化算法——蚁群算法(Ant Colony Optimization, ACO),并应用该方法解决了一系列组合优化问题,取得了较好的实验结果。此后,蚁群算法逐渐被研究者们应用到实际工程问题中。

2.2 基于蚁群算法的非结构化 P2P 网络搜索算法的设计与分析

在基于蚁群算法机制的搜索算法中,查询消息包可以看作是蚂蚁,搜索的目标视为食物,存在搜索目标的节点就是食物源。当源节点发出搜索请求时,就相当于派出蚂蚁到网络中寻找食物。网络中的节点都维护一张信息素表,当蚂蚁到达时,先看节点是否有需要的食物,如果有就返回一个命中消息包(可看作派出一只找到食物的蚂蚁沿原路返回源节点,沿途释放信息素,即修改节点的信息素表)。负责搜索食物的蚂蚁,根据节点中的信息素浓度,选择离目标近的邻居继续爬行,直到 TTL(存活时间)减为 0。通过蚂蚁释放信息素的这样的正反馈的机制,可以实现高效的搜索。

同文献[5],文中假设网络中每个对等节点有两个不同的概要文件(profile),分别是 P_I 和 P_S 。 P_I 是信息概要文件(the informational profile),表示该节点共享给其他节点的文件的信息; P_S 是搜索概要文件(the search profile),表示该节点用户感兴趣的信息(即想要搜索得到的信息)。如果有 1024 个不同的概要文件,可以用 10 位二进制数来标记。查询消息包 M 也是一个 10 位的二进制数。

网络中每个节点维护一张信息素表,节点的第 j 个邻居对应查询内容有一个信息素量 τ_j ,设其初始值为 0。 τ_j 根据下式进行更新:

$$\tau_j = (1 - \rho) \cdot \tau_j + \Delta\tau_j \quad (1)$$

其中, ρ 为信息素的蒸发率,通常设置 $\rho < 1$,以避免信息素量的无限累加; $\Delta\tau_j$ 表示信息素增量。当有命中消息包从某个邻居返回时,节点对该邻居相应的信息素进行本地更新, $\Delta\tau_j$ 就是找到食物的蚂蚁留在节点中的信息素量:

$$\Delta\tau_j = 10 \cdot \text{TTL} \quad (2)$$

这里, TTL 为命中消息包到达节点时存活时间的值,因此,离目标越近的节点,其信息素量越大。也就是说,蚂蚁在较短路径上会遗留更多的信息素,为后来的寻找食物的蚂蚁提供方便。每隔一段时间,信息素会自动更新,形成信息素的挥发现象,这时, $\Delta\tau_j$ 取值为 0。

在搜索过程中,蚂蚁到达某节点时,使用信息素浓度来计算向邻居 j 爬行的概率:

$$p_j = \frac{\tau_j}{\sum_{k \in K} \tau_k} \quad (3)$$

其中 K 为该节点的邻居集合。

消息包在网络中传播,当节点 A 收到一个查询消息包 M 时,先查看自己是否有 M 要找的文件,如果有就返回一个命中消息包;然后,转发消息包给邻居节点。在这个操作中,节点首先挑出概率大于等于 ρ ($\rho < 1$) 的邻居节点来

转发消息包。 N 为节点至少应该转发的消息包数。如果此时转发的消息包的数量没有达到 N , 则节点要继续复制生成消息包来满足要求, 并且以 β 的概率突变每个消息包中的一位, 其目的是通过查找类似的文件来满足搜索 (β 的取值关系到查询的目标文件与类似文件的相似程度, 一般取值为 0.1); 然后, 从余下的概率小于 p 的邻居中随机挑选节点来发送消息包。算法形式描述如下:

算法 1 基于蚁群算法的非结构化 P2P 网络搜索算法

输入: 消息包 M

输出: 返回命中消息包或者转发消息包给选中的邻居节点

```

1) Begin
2)  If  $M$  是查询消息包 then
3)  {
4)  If ( $P_1 = M$ ) then 生成一个命中消息包按原路返回;
5)   $N_{\text{new}} = 0$ ; //  $N_{\text{new}}$  为节点要转发的消息包
6)  For 节点 A 的每个邻居  $j$  do
7)  {
8)      计算转发消息包的概率  $p_j$ ;
9)      If ( $p_j \geq p$ ) then
10)         {
11)             节点复制生成一个消息包并转发给该邻居;
12)              $N_{\text{new}} = N_{\text{new}} + 1$ ;
13)         }
14)     }
15)  If ( $N_{\text{new}} < N$ ) then //  $N$  为节点至少要转发的消息包数量
16)     {
17)         复制生成  $N - N_{\text{new}}$  个消息包;
18)         // 下面是查询类似文件来增加满足消息包的机会
19)         以  $\beta$  的概率突变每个消息包中的一位;
20)         从  $p_j < p$  的邻居中随机挑选  $N - N_{\text{new}}$  个来发送消息包;
21)     }
22)     {
23)         If  $M$  是命中消息包 then
24)             {
25)                 更新与返回邻居对应的信息素;
26)                 按原路返回命中消息包;
27)             }
28) End

```

上述搜索算法将搜索行为模拟成蚂蚁觅食行为, 即搜索开始时, 蚂蚁随机地搜索网络, 随着时间的流逝, 蚂蚁将找到从源节点到食物源的路径, 并在沿途留下信息素。在后来的搜索过程中, 节点将根据信息素浓度, 派出多数蚂

蚁搜索可能存在食物的空间, 而只用少数蚂蚁搜索未知的空间, 从而得到更好的搜索结果。此算法的优点在于, 利用搜索的正反馈结果, 只需要局部知识, 就可以有效地指导后来搜索的前进方向。在搜索开始时, 算法的搜索行为表现为随机宽度优先搜索 (Random Breadth-First Search, RBFS)^[9], 蚂蚁在网络中是随机地爬行。经过几次搜索之后, 由于离查询目标越近, 其信息素浓度越大, 从而使得后来的蚂蚁总是沿着较短的路径前进, 最终形成一条最短路径。

3 结 语

蚁群算法是一种基于蚂蚁种群的新型优化方法, 它已经解决了一系列组合优化的问题, 并可以应用于多代理系统研究等领域。文中将蚁群算法的思想引入非结构化 P2P 网络的搜索算法中, 模拟蚂蚁觅食行为来查找用户需要的文件, 利用蚂蚁的信息素轨迹来指导搜索的前进方向。这种正反馈机制使得搜索可以尽快地找到目标, 得到更好的搜索结果。

参考文献:

- [1] Zeinalipour - Yazti D, Kalogeraki V, Gunopulos D. Information Retrieval Techniques for Peer - to - Peer Networks[J]. Computing in Science & Engineering, 2004, 6(4): 20 - 26.
- [2] 吕建明, 刘悦, 丁林, 等. P2P 与信息检索[J]. 信息技术快报, 2005, 3(2): 10 - 20.
- [3] Lv Q, Cao P, Cohen E, et al. Search and Replication in Unstructured Peer - to - Peer Networks[A]. In: Proc of the 16th international conference on Supercomputing (ICS'02) [C]. New York: ACM Press, 2002. 84 - 95.
- [4] Lin Tsungnan, Wang Hsinping, Wang Jianming. Search Performance Analysis and Robust Search Algorithm in Unstructured Peer - to - Peer Networks[A]. In: Proc of 2004 IEEE International Symposium on Cluster Computing and the Grid (CCGrid 2004) [C]. Piscataway, NJ: IEEE Press, 2004. 346 - 354.
- [5] Ganguly N, Deutsch A. Developing Efficient Search Algorithms for P2P Networks Using Proliferation and Mutation [A]. In: Proc of ICARIS 2004, LNCS 3239 [C]. Berlin, Heidelberg: Springer - Verlag, 2004. 357 - 371.
- [6] Babaoglu Ö, Meling H, Montresor A. Anthill: A Framework for the Development of Agent - Based Peer - to - Peer Systems [A]. In: Proc of the 22nd International Conference on Distributed Computing Systems (ICDCS'02) [C]. Los Alamitos, CA: IEEE Computer Society Press, 2002. 15 - 22.
- [7] Michlmayr E, Graf S, Siberski W, et al. Query Routing with Ants [A]. In: Proc of the Workshop on Ontologies in Peer - to - Peer Communities, European Semantic Web Conference [C]. Heraklion, Greece: [s. n.], 2005. 35 - 46.
- [8] 李士勇. 蚁群算法及其应用 [M]. 哈尔滨: 哈尔滨工业大

由此可见,运用闭操作, $f_E(i, j)$ 与 $f(i, j)$ 中汉字的宽度基本没有改变,但是使扫描进入的手写体汉字的轮廓线更为光滑,消弥了汉字笔划之中狭窄的间断和长细的鸿沟,消除了小的孔洞,并填补了轮廓线中的断裂。

3.2 细化方法

文中所用的细化方法^[10]是通过使用 3×3 结构元素 $\{B\} = \{B^1, B^2, \dots, B^n\}$ 模板序列对图像 $f_E(i, j)$ 中的每一点闭运算操作。通过迭代依次剥去笔划区域边界,直至区域被细化成一条线。对于每一个黑像素点,如果没有上邻点(或下邻点 \ 左邻点 \ 右邻点),且不是孤立点或终止线,去除该黑像素点不会使区域断开,则去除该黑像素点;重复这一步骤直到没有黑像素点可以去除。定义细化后的汉字图像为 $f_F(i, j)$,公式如下:

$$f_F = f_E \circ \{B\} = ((\dots((f_E \circ B^1) \circ B^2) \dots) \circ B^n) \quad (6)$$

使用结构元素 $\{B\} = \{B^1, B^2, \dots, B^n\}$ 进行细化,细化操作表示 $f_E \circ \{B\}$ 。结构元素 B 选用如下模板:

<table><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>*</td><td>1</td><td>*</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table> B^1	0	0	0	*	1	*	1	1	1	<table><tr><td>*</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>*</td></tr></table> B^2	*	0	0	1	1	0	1	1	*	<table><tr><td>1</td><td>*</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>*</td><td>0</td></tr></table> B^3	1	*	0	1	1	0	1	*	0	<table><tr><td>1</td><td>1</td><td>*</td></tr><tr><td>1</td><td>1</td><td>0</td></tr><tr><td>*</td><td>0</td><td>0</td></tr></table> B^4	1	1	*	1	1	0	*	0	0
0	0	0																																					
*	1	*																																					
1	1	1																																					
*	0	0																																					
1	1	0																																					
1	1	*																																					
1	*	0																																					
1	1	0																																					
1	*	0																																					
1	1	*																																					
1	1	0																																					
*	0	0																																					
<table><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>*</td><td>1</td><td>*</td></tr><tr><td>0</td><td>0</td><td>0</td></tr></table> B^5	1	1	1	*	1	*	0	0	0	<table><tr><td>*</td><td>1</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>*</td></tr></table> B^6	*	1	1	0	1	1	0	0	*	<table><tr><td>0</td><td>*</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>*</td><td>1</td></tr></table> B^7	0	*	1	0	1	1	0	*	1	<table><tr><td>0</td><td>0</td><td>*</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>*</td><td>1</td><td>1</td></tr></table> B^8	0	0	*	0	1	1	*	1	1
1	1	1																																					
*	1	*																																					
0	0	0																																					
*	1	1																																					
0	1	1																																					
0	0	*																																					
0	*	1																																					
0	1	1																																					
0	*	1																																					
0	0	*																																					
0	1	1																																					
*	1	1																																					

图像细化后提取出了汉字的骨骼,减少冗余的信息量,突出了汉字的形状结构特征,保证了手写体汉字笔划宽度归一化准确性。

3.3 改进的膨胀运算

膨胀采用如图4所示的大小为 $n \times n$ 的结构元素 C 对细化后的汉字图像 $f_F(i, j)$ 进行处理,其中 $n = wh$ 。经过膨胀运算后,输入的手写体汉字样本的笔划宽度按照所希望的有效归一化为前面所提取的笔划宽度 wh 。

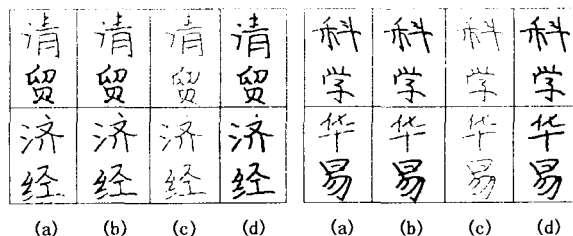
	1	2	...	n
1	1	1	...	1
2	1	1	...	1
\vdots	\vdots	\vdots	\vdots	\vdots
n	1	1	...	1

图4 结构元素 C

3.4 笔划宽度归一化实验结果

实验结果如图5所示。图中(a)为手写体汉字原图,可根据(3)式中的阈值对手写体汉字图像进行二值化处理,然后再进行闭运算得到如图(b)所示的手写体汉字图像。取已经过闭运算的手写体汉字图像(b),对其进行细化后结果如图(c)所示。对细化后的手写体汉字图像(c)

进行膨胀运算,最终可得到笔划宽度归一化为同一宽度 wh 后的手写体汉字,如图(d)所示。



(a) 汉字原图像; (b) 闭运算后的汉字图像;
(c) 细化后的汉字图像;
(d) 膨胀后的汉字图像(即笔划宽度归一化后的结果)

图5 实验结果

4 结 论

对手写体汉字笔划宽度的提取及笔划宽度的归一化方法进行了研究,提出了一种手写体汉字识别预处理方法,并给出了算法。经实验表明,该方法简单,效果明显。笔划宽度归一化后的手写体汉字能保证原汉字的形状结构特征基本不变。

参考文献:

- [1] 张世辉. 汉字图像预处理算法的研究及实现[J]. 微机发展, 2003, 13(4): 55-58.
- [2] 金连文, 黄建成, 尹俊勋. 变形变换及其在汉字识别中的应用[J]. 中国图象图形学报, 2002, 7(2): 170-175.
- [3] 陈友斌, 丁晓青, 吴佑寿. 一种新的用于汉字识别的非线性归一化方法[J]. 模式识别与人工智能, 1998, 11(3): 310-317.
- [4] 杨 静. 基于数学形态学的图像分割研究及应用[D]. 合肥: 合肥工业大学自动化智能研究所, 2005.
- [5] 高 学, 金连文, 尹俊勋. 基于笔划密度的弹性网格特征提取方法[J]. 模式识别与人工智能, 2002, 15(3): 351-354.
- [6] 徐 睿, 金连文, 高 学, 等. 基于数学形态学的手写体汉字特征提取方法[A]. 第十二届全国神经网络学术年会[C]. 北京: [出版者不详], 2002. 470-476.
- [7] 金连文, 徐 睿, 龙钧宇. 一种简单有效的手写体汉字笔划粗细归一化方法[DB/OL]. 中国科技论文在线. <http://www.paper.edu.cn/>, 2003.
- [8] Sonka M, Hlavac V, Boyle R. 图像处理、分析与机器视觉(第2版)[M]. 北京: 人民邮电出版社, 2003.
- [9] 向世明. Visual C++ 数字图像与图形处理[M]. 北京: 电子工业出版社, 2002.
- [10] 王建平, 钱自拓. 基于数学形态学的图像汉字笔划细化和提取[R]. 合肥: 合肥工业大学自动化智能研究所, 2005.

(上接第 28 页)

- 学出版社, 2004.
- [9] Kalogeraki V, Gunopulos D, Zeinalipour-Yazti D. A Local Search Mechanism for Peer-to-Peer Networks[A]. In:

Proc of 11th Int'l Conf Information and Knowledge Management (CIKM'02)[C]. New York: ACM Press, 2002. 300-307.