

# 禁忌搜索算法用于解决网络路由问题

王东平<sup>1,2</sup> 李绍荣<sup>2</sup>

(四川大学电气信息学院 成都610065)<sup>1</sup> (电子科技大学电子工程学院电子系统工程研究所 成都610054)<sup>2</sup>

## Tabu Search Technique for Solving the Routing Problem

WANG Dong-Ping<sup>1,2</sup> LI Shao-Rong<sup>2</sup>

(College of Electric Information, Sichuan University, Chengdu 610065)<sup>1</sup>

(Institute of Electron System Engineering, College of Electron Engineering, UESTC, Chengdu 610054)<sup>2</sup>

**Abstract** Routing problem is a very important problem in the network design. However, with the increasing of the number of vertices, the convergence speed of the conventional method (such as the Dijkstra algorithm) becomes slow. In some services, the accurate shortest path isn't requested. This paper presents a new algorithm for solving this problem based on the tabu search technique. The tabu search algorithm can get the satisfied path with the changing of the iteration times, the tabu period and neighborhood size. Simulation results demonstrate that the proposed method is very efficient for computing the shortest path, especially when the scale of the network is large.

**Keywords** Routing, Optimization, Tabu search

## 1. 引言

在网络设计中,经常会遇到这样一类问题:在由多个路由器组成的网络中,路由器之间的各个直接链路都具有各自的传输时延,求解从某一个路由器出发,到达另一个路由器的最短时延,这就是路由选择问题。从数学的角度可简化为:在由多个节点组成的网络中,节点与节点之间的链路具有各自的距离,求解一条从某一节点到达另一节点的最短距离的路径<sup>[1]</sup>。

利用传统的方法(如 Dijkstra 算法)来解决该问题往往会随着网络节点数的增加,收敛速度明显变慢,为此人们不断地研究出了一些新的算法。文[2]提出利用拉格朗日松弛因子法与次梯度寻优法来找到最优解的范围和可行解,但算法复杂,计算和编程工作量大。文[3]将其近似为一个二次型优化问题,利用一种具有全局收敛性质的神经网络给予解决。文[4]提出利用遗传算法解决这一问题。

本文提出了一种建立在禁忌搜索算法基础上的新的路由选择算法。可以通过调整算法的迭代次数、禁忌表的长度以及候选集合的长度,得出满足要求的解。仿真结果表明,这是一种有效的计算最短路的方法,尤其是对于大型的稠密网络。在本文中我们利用邻接矩阵来表示网络。

## 2. 禁忌搜索算法

禁忌搜索(Tabu Search)算法是一种启发式搜索算法<sup>[5,6]</sup>,它和一些典型的局部搜索算法不同,它采用了禁忌技术,可以避免在某个邻域内重复搜索而陷入局部极小,它可以允许向一个使目标函数变坏的方向移动,只要该移动有获得更好解的潜力。Tabu Search 算法是继模拟退火算法和遗传算法之后出现的又一种十分有效的全局搜索方法。Tabu Search 算法已在诸多组合优化领域表现出强大的寻优能力,并以其较高的求解质量和效率得到人们越来越多的青睐。对一些复杂的优化问题,这一方法显示出极强的寻优能力,用该方法得到的解在很多问题中超过了以往用其它方法得到的

最好解<sup>[7~10]</sup>。

在禁忌搜索算法中有如下一些基本元素需要定义:

- 移动( $s' \rightarrow s''$ ):从一个解( $s'$ )到另一个解( $s''$ )的变迁。
- 邻域(备选解集):当前解的所有可能移动的集合。
- 禁忌约束:有一些限制条件使得某些移动是被禁止的,这些被禁止的移动被称为被禁忌的。被禁忌的移动被放在一个表中,这个表被称为禁忌表。经过一段时间后(禁忌表长或禁忌周期),这些移动将被释放出来可以重新访问。
- 特赦条件(期望准则):有一些规则凌驾于禁忌约束之上,当这些规则满足时,即使一些移动是被禁忌的,也可以使这些移动被采用。

给定上述基本元素,禁忌搜索算法可以被描述如下:从一个初始解出发,估计该解的目标函数值,然后选定备选解集。如果最好的移动不是被禁忌的,或者是被禁忌的但满足特赦条件,那么就选择该移动,并把该备选解作为新的当前解;否则,选择不被禁忌的最好移动所对应的备选解作为当前解。重复以上过程,终止时得到的最好解就作为用禁忌搜索算法解决该问题的最终解。

用禁忌搜索算法解决优化问题的流程如下:

1. 初始化 产生初始解,并初始化在禁忌搜索过程中需要用的参数。
2. 循环 产生备选解集(邻域),计算各个备选解的目标函数值,根据目标函数值、禁忌状态和特赦条件选择一个备选解,从当前解移动到该备选解,作为新的当前解,将新的当前解放入禁忌表中,使其保持一个禁忌周期的禁忌状态。
3. 停止 直到预先设定的停止条件满足。

## 3. 禁忌搜索算法应用于路由问题

禁忌算法应用于路由选择时各个主要部分描述如下。

1. 解的编码。我们用一条路径所经过的节点组成的向量来表示一个解。例如我们用向量[1 3 5 6]来表示依次经过节点1,节点3,节点5,节点6的一条路。
2. 初始解的选取。在网络图中随机地选择一条从起始节

点到目的节点的通路。

3. 邻域选择。在初始解向量中随机选择某个节点*i*，并查找初始向量解中由*i*开始的第三个节点*j*，找到另外一条由*i*到*j*的路由，替换掉初始向量解中原来从*i*到*j*的部分，做为一个备选解，反复5次，找出5个备选解。在具体设计的邻域算法中，我们要考虑如何防止找到的从*i*到*j*的路由中，含初始解中去除我们要替换的节点集合以外的节点。我们采用的方法是通过改变网络矩阵中某些行列的值，来达到这个目的。假如初始路由为[1 2 3 4 5 6 7 8 9]，当我们要替换从4到7的一段时，我们要把表示这个图的邻接矩阵的1、2、3、8、9行和列全部置0，用处理后的矩阵所表示的网络中找替换路由，就可避免组成的新路中重复走过同一个节点。

4. 评价函数的设计。我们用一条路所经过的节点之间的弧的权值的和函数做为评价函数。

5. 为了简便起见，我们没有使用特赦规则。如果在上一步中找到的解发现在禁忌表中，则重新开始找另外不同的备选解。

6. 终止规则是迭代次数*N*次或连续*M*次迭代没有改进。

结合以上各点及上一节中禁忌搜索算法的流程，我们就可以得到用禁忌搜索解决路由选择这个优化问题的算法。

#### 4. 计算机仿真

在本节中我们通过两个例子来说明所提出的算法的有效性，为了便于比较，我们也对同样的例子用 Dijkstra 算法进行求解。

例1 网络图由图1所示，它是一个具有10个节点的网络，其邻接矩阵是

$$A = \begin{bmatrix} 0 & 4 & 0 & 6 & 3 & 8 & 0 & 0 & 0 & 0 \\ 4 & 0 & 7 & 0 & 0 & 0 & 0 & 18 & 0 & 0 \\ 0 & 7 & 0 & 4 & 15 & 0 & 4 & 0 & 17 & 0 \\ 6 & 0 & 4 & 0 & 1 & 0 & 11 & 14 & 0 & 0 \\ 3 & 0 & 15 & 1 & 0 & 7 & 0 & 0 & 0 & 0 \\ 8 & 0 & 0 & 0 & 7 & 0 & 5 & 5 & 0 & 0 \\ 0 & 0 & 4 & 11 & 0 & 5 & 0 & 0 & 8 & 2 \\ 0 & 18 & 0 & 14 & 0 & 5 & 0 & 0 & 0 & 8 \\ 0 & 0 & 17 & 0 & 0 & 0 & 8 & 0 & 0 & 6 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 8 & 6 & 0 \end{bmatrix}$$

我们先用 Dijkstra 算法找从节点 B 到节点 J 的最短路，得到结果为 B C G J，最短路长为13。再用本文的方法计算，终止条件是迭代100次，得到结果为 B C G J，路长为13，是最短路。

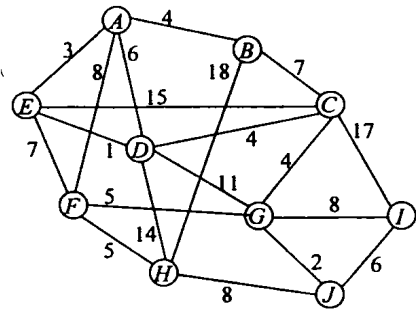


图1 具有10个节点的网络图

例2 网络图由图2所示<sup>[4]</sup>，它是一个具有20个节点的网络图，它的邻接矩阵是一个20×20的对称方阵，因为篇幅所限，在此略去。

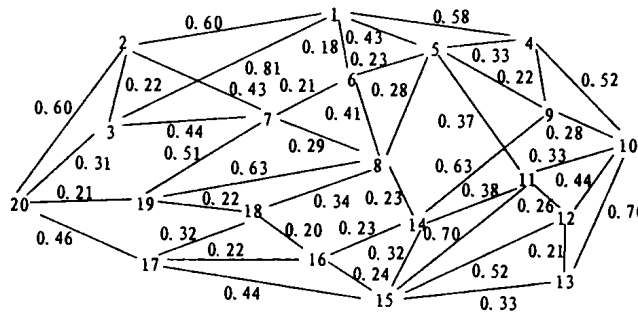


图2 具有20个节点的网络图

先用 Dijkstra 算法找从节点2到节点15的最短路，得到结果为2 7 8 14 15，最短路长为1.27。再用本文的方法计算，终止条件是迭代200次或最大没有改善连续迭代次数为15，得到结果是2 7 8 14 15，路长为1.27，也找到了最短路。

我们对此例用 Dijkstra 算法和本文中提出的算法分别重复计算20次。用本文提出的算法有17次得到最优解，用 Dijkstra 算法和我们提出算法的平均用时比为5.8:1。可见在本例这种网络中用我们提出的方法可以用更少的时间得到满意的结果。

结论 本文我们提出了一种利用禁忌搜索技术进行路由选择的算法。我们详细地给出了该方法的具体实现细节。该方法的有效性通过计算机仿真来加以说明，从仿真结果可以看出，此方法在并不要求精确最优的应用场合下起到较好的效果，同时节省大量的处理器时间，特别是在大型稠密网络中，其优势更加明显，不失为一种路由选择算法的新的有效方法。

致谢 感谢孙大伟先生有益的讨论和所做的仿真工作。

#### 参考文献

- 1 Tanenbaum A S. 计算机网络. 熊桂喜, 王小虎译, 第三版. 北京: 清华大学出版社, 1998
- 2 Gavish B, Hantler S L. An algorithm for optimal route selection in SNA networks. IEEE Trans. on Communications, 1983, 31 (10): 1154~1167
- 3 张顺颐, 邵艾育, 沈苏彬, 吴新余. 一种基于神经网络模型的计算机通信网络延迟和流量分配新算法. 通信学报, 1995, 16(6): 1~8
- 4 Leung Y, Li G, Xu Z B. A genetic algorithm for the multiple destination routing problems. IEEE Trans. on Evolutionary Computation, 1998, 2(4)
- 5 Glover F. Tabu search, part I. ORSA J. Comp., 1989, 1: 190~206
- 6 Glover F. Tabu search, part I. ORSA J. Comp., 1990, 2: 4~32

# 可伸缩 TAGS

闵帆 张君雁 杨国纬  
(电子科技大学计算机学院 成都610054)

## Scalable TAGS

MIN Fan ZHANG Jun-Yan YANG Guo-Wei  
(Computer Science Institute, UESTC, Chengdu 610054)

**Abstract** In a distributed Web server system where tasks are unpreemptible, the most important issue for improving quality of service (QoS) is how to realize fairness and reduce average slow down. In this paper we present an algorithm named Scalable TAGS by integrating Central Queue algorithm and Task Assignment by Guessing Size (TAGS), together with its performance analysis, system parameter setting algorithm subject to fairness requirement, and optimal grouping method.

**Keywords** Distributed web server system, Task assignment, Fairness, Slow down, Bounded pareto

## 1 引言

近年来,分布式 Web 服务器系统由于其价廉、易于扩充等特点被广泛采用。该类系统的模型如图1所示。任务到达任务分配服务器后,被分配到某一个任务处理服务器(下文中均简称为服务器)执行。在该类系统中的任务分配策略对服务质量起着决定性作用。

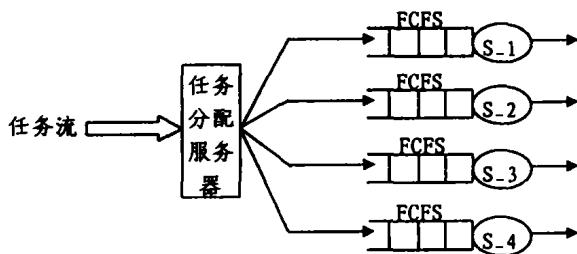


图1 任务分配模型

若把本文假设条件(见2.1节)中的任务长度改成指数分布,则集中队列算法能获得最佳性能<sup>[1]</sup>;但任务长度服从有界 Pareto 分布,且服务器数量较少时, TAGS 算法能获得比集中队列算法好得多的性能<sup>[2]</sup>。

本文结合集中队列算法与 TAGS 算法,提出一种可伸缩的 TAGS 算法,并依次给出其性能计算式、公平性要求下的系统参数设置算法及服务器最佳分组方法。

## 2 可伸缩 TAGS 算法

### 2.1 假设条件

本文采用与文[2]相同的假设条件:1. 任务长度(按处理

所需时间计)不可预知;2. 任务不可剥夺,即任务在执行过程中被中断,则已做工作全部作废;3. 各服务器以先来先服务的方式处理任务;4. 任务到达服从 Poisson 分布;5. 任务长度服从有界 Pareto 分布;6. 各服务器处理能力完全相同。

### 2.2 性能评价标准

定义1(延迟, delay) 任务从进入系统到正式被处理的间隔时间。

定义2(延迟比, slow down) 任务延迟与其处理时间需求之比。

定义3(服务器平均延迟比) 在指定服务器完成任务的延迟比均值。

定义4(系统平均延迟比) 系统内所有任务的延迟比均值。

定义5(任务级公平) 系统中所有任务具有相同的延迟比。

定义6(服务器级公平) 系统中所有服务器具有相同的平均延迟比。

在多数情况下,定义5中所描述的任务级公平难以达到;我们采用定义6所描述的服务器级公平作为前提,并以定义4所描述的系统平均延迟比为性能评价标准。

### 2.3 算法描述

可伸缩 TAGS 算法如图2所示。

任务到达任务分配服务器时,若  $S_{i-1}$  (图中  $i=1, 2, 3$ ) 中有一个或多个空闲,则将该任务分配给其中最早进入空闲状态者,否则将任务放入任务分配服务器队尾;  $S_{i-1}$  完成当前任务时,若任务分配服务器中队列不为空,则取出其队首任务执行,否则进入空闲状态。

到达  $S_i$  的任务如果在  $s_i$  (具体说明见表1) 个单位时间

闵帆 博士研究生,主要研究方向:分布式系统、主动网络。张君雁 博士研究生,主要研究方向:移动 Agent、主动网络。杨国纬 教授,博士生导师,主要研究方向:计算机网络、数据库。

博士研究生,主要研究方向:移动 Agent、主动网络。杨国纬 教授,博士

7 Cvijovic D, Klinowski J. Taboo search: an approach to the multiple minima problem. *Science*, 1995, 267: 664~666

8 Laguna M, Glover F. Bandwidth packing: a tabu search approach. *Management Science*, 1993, 39: 492~500

9 Lee C Y, Kang H G. Cell planning with capacity expansion in mo-

bile communications: a tabu search approach. *IEEE Trans. on Veh. Tech.*, 2000, 49: 1678~1691

10 Matsumura T, Nakamura M, Tamaki S, Onaga K. A parallel tabu search based on aspiration control and its cooperative execution. *IEICE Trans. on Fundamentals*, 2000, E83-A: 2196~2202