

基于改进蚁群算法的计算机网络路由优化研究

叶华乔

(武汉船舶职业技术学院 湖北 武汉 430050)

摘要: 计算机网络规模的逐渐扩大使数据传输时的延时、丢包等现象日益明显。为了提高网络数据传输的稳定性,降低网络消耗,研究使用蚁群算法解决计算机网络的路由优化问题。同时,为了提高蚁群算法的性能,提出了状态转移规则和信息素更新规则的改进策略,使蚁群算法的收敛速度得到明显提升。仿真结果表明,上述改进蚁群算法可以在较短时间内计算出路由优化的结果,优化成功率较高,非常适合实际应用。

关键词: 路由优化; 计算机网络; 蚁群算法

中图分类号: TP393.02 **文献标识码:** B

Research on Routing Optimization of Computer Network Based on Improved Ant Colony Algorithm

YE Hua-qiao

(Wuhan Marine Vocational and Technical College, Wuhan Hubei 430050, China)

ABSTRACT: The problem of data transmission delay and packet loss is becoming more evident with the gradually expand of the computer network scale. In order to improve the stability of the network data transmission and reduce network consumption, the ant colony algorithm was studied to solve computer network routing optimization problem. Meanwhile, in order to improve the performance of the ant colony algorithm, the improvement strategies of the state transition rule and pheromone updating rule were proposed, which makes the speed of convergence enhanced significantly. Simulation results show that the proposed improved ant colony algorithm can obtain route optimization results in a relatively short period of time with high success rate, and very is suitable for practical application.

KEYWORDS: Routing optimization; Computer network; Ant colony algorithm

1 引言

随着科学技术的进步,互联网等新兴事物取得了长足的发展,在人们日常生活中的地位越来越重要。在享受计算机网络带来便利的同时,许多专家学者开始关注网络路由的优化问题。由于现代网络的规模日益扩大,在传输数据时经常出现延时、丢包、带宽限制等问题,严重影响了网络的正常运作。因此,需要研究一种优良的策略,来有效的管理和利用网络资源^[1-2]。

计算机网络的路由优化问题,实质上就是一类特殊的NP-hard组合优化问题。这类优化问题属于难以求解的组合最优化问题,由于这类问题的应用领域较多,因此受到了非常多的关注。传统的优化算法一般是基于统计数学原理,需要在精确已知或大部分已知数学模型的前提下,才能够较

好的解决其优化问题。但是在实际的应用中,很多问题的数学模型非常复杂,涉及多个学科的交叉应用,往往无法精确获得,尤其是在约束条件较多的情况下,优化问题的最优解很难得到,即使可以求解,也需要较长时间,因此传统算法并不适合解决现代大型计算机网络的路由优化问题^[3]。随着NP-hard理论的出现和进步,许多新兴的智能算法,例如神经网络、遗传算法以及蚁群算法等,开始被应用到网络路由优化之中^[4-6],取得了一定的效果,但是或多或少都存在一些问题,比如神经网络容易陷入局部极小、遗传算法容易早熟收敛等。

基于上述分析,本文研究利用蚁群算法来解决计算机网络的路由优化问题。文中对路由优化的数学实质进行了详细阐述,同时提出了基本蚁群算法在状态转移规则和信息素更新规则两方面的改进措施,较好地解决了蚁群算法收敛速度慢,容易陷入局部极小值的问题。仿真实验表明,本文提出的优化方法效果较好,具有一定的实用性。

收稿日期: 2014-11-05 修回日期: 2014-12-17

2 网络路由优化问题描述

所谓网络路由优化问题,其实就是指在一个规模已知的计算机网络中,在满足某一个或某几个约束条件下的前提下,找到从路由节点 A 到路由节点 B 之间的最佳路径^[7]。计算机网络模型可以用一个无向赋权图 $G=(V,E)$ 来描述,其中 V 代表节点集合, E 代表链路集合,规定相邻节点之间最多只能有一条有效链路。本文重点研究的是单播路由优化,即在给定的源节点和目标节点之间,找到一条满足延时、丢包率、带宽等条件,同时消耗又最小的最优路径。

对于计算机网络中的任意一条链路,一般会有 4 种影响数据传输的因素:延时、延时抖动、带宽、链路消耗。延时主要是指数据包在网络中传输的平均用时;延时抖动是指数据包在网络中传输时间的长短变化;带宽是指网络链路所能承受的最大数据通信量;链路消耗主要指能量、费用等。对于计算机网络中的节点来说,也存在三种因素,包括延时、延时抖动和丢包率。其中,延时和延时抖动的定义与上述相同,丢包率主要是指数据在网络传输过程中损坏或丢失的概率。因此,网络路由优化问题可以归结为:

目标函数: $\min cost = cost(e)$ e 是通信链路

约束条件:

$$1) \text{ 延时约束: } \sum_{e \in EL} delay(e) + \sum_{n \in VL} delay(n) \leq D$$

2) 延时抖动约束:

$$\sum_{e \in EL} delay_jitter(e) + \sum_{n \in VL} delay_jitter(n) \leq DJ$$

$$3) \text{ 带宽约束: } bandwidth(e) \geq B$$

$$4) \text{ 丢包率约束: } packet_loss(n) \leq PL$$

其中 D 、 DJ 、 B 和 PL 分别代表 QoS 意义下的延时、延时抖动、带宽和丢包率约束, EL 和 VL 分别是链路集合和节点集合。

3 蚁群算法及其改进

3.1 蚁群算法的基本原理

1991 年,意大利著名学者 M. Dorigo 等人提出了一种智能的启发式进化算法,用于解决组合优化问题,该算法的主要思想是模拟自然界中蚁群协同合作觅食的行为,因此被成为蚁群算法^[8]。根据蚂蚁觅食的特性,它们会在寻找的路径上留下信息素,路径越短,蚂蚁来回所用的时间越短,留下的信息素就会相应增多,其它蚂蚁会被自觉的从其它路径上吸引过来,最终找到一条完成任务所需的最短路径。假设蚂蚁 k 在 t 时刻从节点 i 转移到节点 j ,其状态转移概率用如下等式描述

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \times [\eta_{ij}(t)]^\beta}{\sum_{s \in allowed_k} [\tau_{is}(t)]^\alpha \times [\eta_{is}(t)]^\beta} & j \in allowed_k \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

其中 $\tau_{ij}(t)$ 为节点 i 到节点 j 这条路径上 t 时刻的信息强度,

在开始阶段每条路径上的信息强度相等; $\eta_{ij}(t)$ 为启发因子,它是路径长度倒数的函数,即 $\eta_{ij}(t) = 1/l_{ij}$; $allowed_k$ 为蚂蚁 k 在第 t 次迭代中可选的节点集;参数 α 和 β 分别表示信息素和距离在选择概率上的作用。

蚂蚁在寻找路径的过程中会不断留下信息素,当信息素累积过多时,启发信息就会受到影响,所以信息素需要不断更新。经典蚁群算法的更新规则为

$$\tau_{ij}(t+n) = (1-\rho) \cdot \tau_{ij}(t) + \Delta\tau_{ij}(t) \quad (2)$$

$$\Delta\tau_{ij}(t) = \sum_{k=1}^m \Delta\tau_{ij}^k(t)$$

其中 $\rho \in (0,1)$ 为信息素保留系数, $\Delta\tau_{ij}(t)$ 为路径 (i,j) 上新增加的信息素总和, $\Delta\tau_{ij}^k(t)$ 为蚂蚁 k 在时刻 $(t+n)$ 之间在此路径上遗留的信息素总和,其计算公式如下

$$\Delta\tau_{ij}^k(t) = \begin{cases} \frac{1}{L_k} & \text{若蚂蚁 } k \text{ 在时刻 } (t+n) \text{ 之间访问此路径} \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

其中 L_k 为蚂蚁 k 行进的所有路径长度之和,其值越小,信息素越多。

从上述蚁群算法的基本原理可以看出,其实质是一种分布式并行算法,在路径规划过程中同时存在正反馈校正机制,这使得其具有较强的求解能力,而且算法的鲁棒性很好。尽管如此,该算法还是存在一些缺陷,例如:算法复杂度过高,对于高维规划问题求解时间太长,收敛速度慢,并且容易陷入局部最优值,因此需要对其进行改进。本文主要研究蚁群算法在状态转移规则和信息素更新规则两方面的改进方法。

3.2 状态转移规则的改进

网络路由优化问题不同于常规的路径规划,数据在路由节点之间传输时存在延时、延时抖动等现象,必需加以考虑。因此,本文用 d 、 dj 以及 c 分别表示延时、延时抖动、带宽和路径能量消耗。在进行节点的状态转移过程中,将延时和延时抖动的影响考虑进去,改进后的状态转移公式如下所示

$$j = \begin{cases} \max_{s \in allowed_k} \{ \tau_{is}(t)^\alpha \times \eta_{is}(t)^\beta \} q \leq q_0 & \\ \text{根据式(1)选择 } j & \text{otherwise} \end{cases} \quad (4)$$

其中 q_0 为 0-1 之间的常数,而 q 为 0-1 之间均匀分布的随机数。启发因子 $\eta_{ij}(t)$ 不再是路径倒数的函数,而是改进为延时和延时抖动和倒数的函数,即 $\eta_{ij} = 1/(d_{ij} + dj_{ij})$ 。经过这样改进以后,可以保证蚂蚁在进行状态转移时,选择信息素强度较高的路径,从而避免陷入局部最优值。

3.3 信息素更新策略的改进

为了进一步提高算法在运行过程中的收敛速度,本文提出了新的信息素更新策略,该策略不仅考虑信息素的局部更新,同时考虑其全局更新。具体来看,在蚂蚁 k 经过路径 (i,j) 时,仍然还是按照式(2)进行局部更新;当蚂蚁 k 完成一次从起点到终点的路径选择以后,按照下式对其所经过的所有路径的信息素进行全局更新

$$\tau_{ij} = (1 - \rho_1) \times \tau_{ij} + \rho_1 \times \frac{1}{Q} \quad (5)$$

对它未经过的路径上的信息素按照下式进行更新:

$$\tau_{ij} = (1 - \rho_1) \times \tau_{ij} \quad (6)$$

其中 ρ_1 仍然为信息素的保留系数, $Q = AQ_1 + BQ_2$, Q_1 表示路径 (i, j) 和节点 i 上的延时和延时抖动之和, Q_2 表示路径 (i, j) 上的能量消耗, B 为总带宽。经过这样的全局更新以后, 非常有利于全局最优解的求得。

3.4 改进蚁群算法的实现步骤

根据本文上述的改进策略, 改进后蚁群算法的实现流程如图 1 所示。

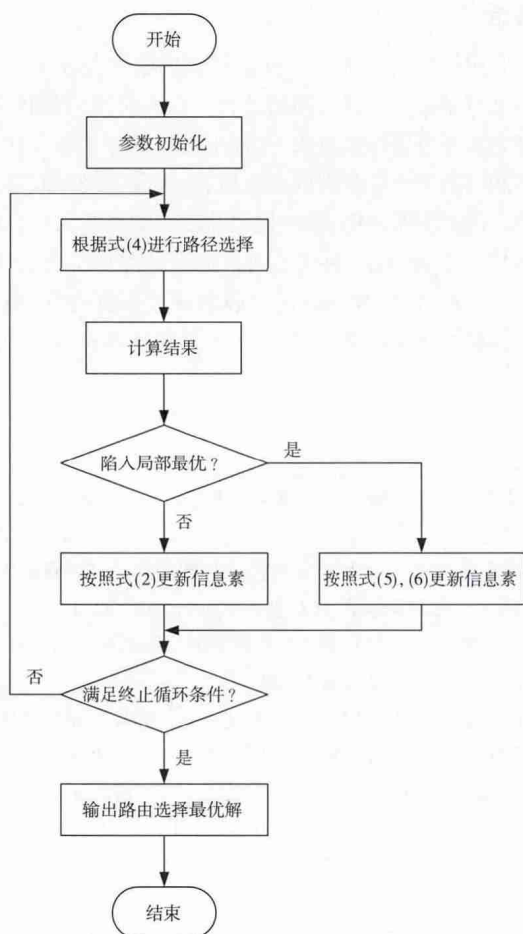


图 1 改进蚁群算法流程图

4 仿真实验

利用 Matlab 软件构造一个小型的拓扑网络, 如图 2 所示。要求实现 3 个单播路由寻优, 分别为 $(1, 6)$, $(2, 6)$ 和 $(3, 8)$, QoS 参数要求为: $B = 70$, $D = 8$, $DJ = 5$, $PL = 0.0001$ 。蚁群算法设定参数为: 蚂蚁 20 只, $\alpha = \beta = 1$, $\rho = \rho_1 = 0.1$, 算法最大迭代次数为 100。

下面根据第 2 节中所述的约束条件 3 和 4, 对原始网络拓扑结构进行裁剪, 由于链路 $(2, 3)$ 和链路 $(4, 8)$ 之间的带宽不满足约束条件 2, 故将它们去掉, 裁剪后的网络拓扑结构

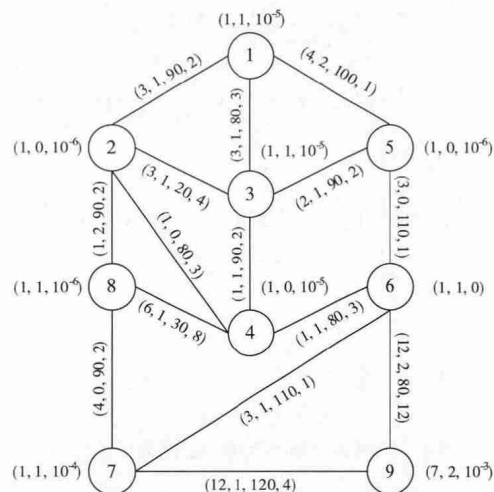


图 2 原始网络拓扑结构

如图 3 所示。链路 $(6, 9)$ 和链路 $(7, 9)$ 之间的丢包率不满足约束条件 3, 也将其去掉, 简化后的网络拓扑结构如图 4 所示。

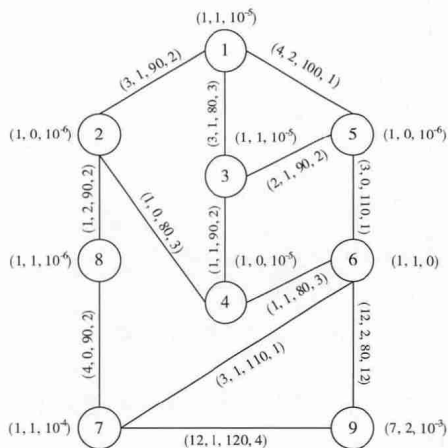


图 3 应用带宽约束条件后网络拓扑结构

路由选择的结果如表 1 所示。路由 $(2, 6)$ 和 $(3, 8)$ 的结果一目了然, 都是正确的, 对于路由请求 $(1, 6)$, 直观来看还有一个费用仅为 2 的结果 $1 \rightarrow 5 \rightarrow 6$, 但是这条路由的延时却达到了 8, 要高于 $1 \rightarrow 2 \rightarrow 4 \rightarrow 6$ 路由的 7, 由于在实际应用中, 延时是优先考虑的因素, 在计算式 (5) 中的 Q 时, 规定了 $A > B$, 因此最终的路由选择结果为 $1 \rightarrow 2 \rightarrow 4 \rightarrow 6$ 。

表 1 路由优化结果

路由请求	路径	费用	延时	延时抖动
$(1, 6)$	$1 \rightarrow 2 \rightarrow 4 \rightarrow 6$	8	7	2
$(2, 6)$	$2 \rightarrow 4 \rightarrow 6$	6	3	1
$(3, 8)$	$3 \rightarrow 4 \rightarrow 2 \rightarrow 8$	7	5	3

下面比较一下改进蚁群算法与经典蚁群算法的性能, 首

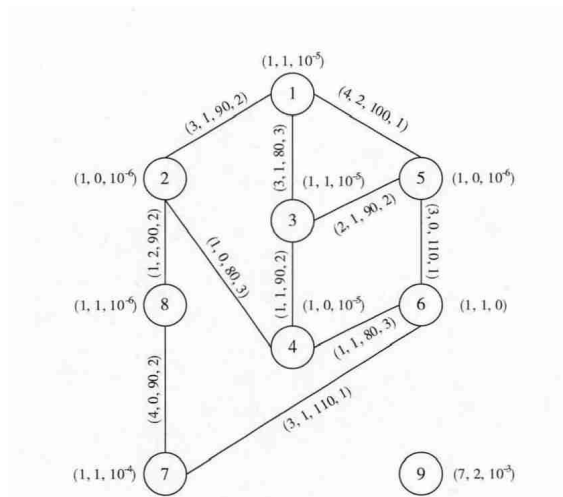


图4 应用丢包率约束条件后网络拓扑结构

先考察收敛速度上的差异,算法运行时间如图5所示。可见改进后的蚁群算法,其加工时间明显小于经典蚁群算法,收敛速度大为提升。

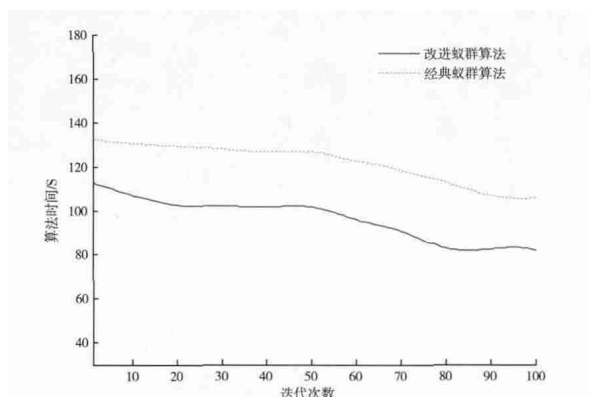


图5 算法收敛速度比较

接下来本文还对路由选择的成功率进行了比较分析。由于蚁群算法属于分布式的并行算法,具有陷入局部最优解的可能,故不是每一次都能成功选择到期望的最优结果。文中选取了 Matlab 软件自带的标准网络测试数据集中的 20 个网络,其中有 USA Network 等 17 个小型网络,还包括 NASA Network、Mini Pacific Ocean Network 以及 Central bank Network 三个中型网络。每个网络都设定 100 个路由路径需要优化,结果如图6所示。

由图6的结果可知,改进后的蚁群算法,其路由优化的成功率要明显高于经典蚁群算法,平均值为 90.25%,而经典蚁群算法仅为 84.45%。尤其是对于 3 个规模较大的中型网络,改进蚁群算法的路由优化成功率分别达到了 93%、92% 和 90%,经典蚁群算法的优化结果均低于 90%。由此可见,在网络规模、参数设置均相同的情况下,改进后的蚁群算法具有非常明显的优势,大大降低了陷入局部最优值的风险,算法整体性能提升明显。

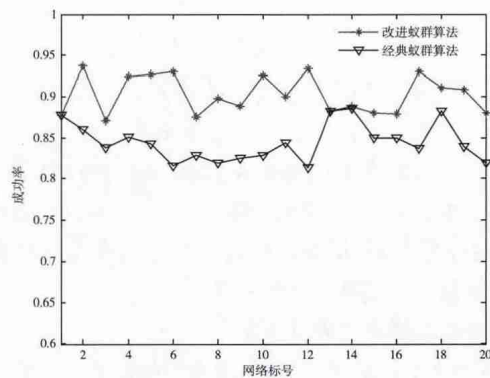


图6 路由优化成功率比较

5 结论

计算机网络路由优化一直是领域内较为关注的一个课题。本文对单播路由优化问题进行了深入研究,利用蚁群算法的分布式并行特性来解决,并提出了蚁群算法节点状态转移规则和信息素更新规则的改进措施,提高了经典蚁群算法的性能。仿真实验表明,本文提出的改进蚁群算法可以根据约束条件的变化,较好的优选出最适合的路由信息。同时,经过两方面的改进,蚁群算法的收敛速度得到提升,陷入局部极值的概率也大为降低,更利于大规模计算机网络的路由优化问题。

参考文献:

- [1] 马伟. 计算机网络路由探究综述[J]. 电子测试, 2013, 13(7): 250-251.
- [2] 高天寒, 郭楠. 一种基于改进 HMIPv6 的移动网络路由优化方案[J]. 东北大学学报(自然科学版), 2012, 33(1): 69-72.
- [3] 任红霞. 基于 CPSO 的无线传感器网络路由优化[J]. 计算机仿真, 2012, 29(3): 202-205.
- [4] 孔玉静, 等. 基于 BP 神经网络的无线传感器网络路由协议的研究[J]. 传感技术学报, 2013, 26(2): 246-250.
- [5] 袁天, 梁俊. 基于遗传算法的卫星 MPLS 网络路由协议研究[J]. 信息工程大学学报, 2013, 14(5): 548-551.
- [6] 史士财, 等. 基于改进蚁群算法的装配序列规划[J]. 计算机集成制造系统, 2010, 16(6): 1189-1194.
- [7] 李祖鹏, 黄建华, 唐辉. 基于 P2P 计算模式的自组织网络路由模型[J]. 软件学报, 2005, 16(5): 916-931.
- [8] M Dorigo, V Maniezzo, A Colomni. The ant system: optimization by a colony of cooperating agents[J]. IEEE Transactions on Systems, 1996, 26(1): 29-41.

【作者简介】



叶华乔(1970-)男(汉族)湖北武汉人,硕士,副高,主要研究方向:计算机应用、数据库技术与算法。