

蚁群算法解决 CSAHLP 问题时的 修正因子的研究

胡 昊, 刘树森, 张小燕, 苏 勇

(江苏科技大学 计算机科学与工程学院, 江苏 镇江 212003)

摘 要: 优化蚁群算法是一种基于种群的模拟进化算法, 其高效的仿生过程在各类组合问题中有了广泛的应用。CSAHLP 经常被用来描述物流在大范围运输时所产生的问题。在 CSAHLP 问题中, 枢流点和节点都是未知参变量, 这使得此问题归类于典型的 NP 问题。ACO 作为高效解决 NP 问题的算法之一, 在 CSAHLP 上有了越来越多的研究应用。但是, 蚁群算法也有其自身缺点, 受容量约束的条件作为外部约束使得蚁群有时无法得出正确的解。文中详细讨论了蚁群产生非可行解的原因及其处理方法, 并通过实验证明方法的有效性。

关键词: 优化蚁群算法; CSAHLP; NP; 非可行解

中图分类号: TP15

文献标识码: A

文章编号: 1673-629X(2012)08-0119-04

Research on Correction Factor in Resolving CSAHLP Through ACO

HU Hao, LIU Shu-sen, ZHANG Xiao-yan, SU Yong

(School of Computer Science and Engineering, Jiangsu University of Science and Technology,
Zhenjiang 212003, China)

Abstract: ACO is a simulation evolutionary algorithm based on population. Its effective bionics process has been widely used in various combinatorial problems. CSAHLP is often used to describe those problems produced when logistics happen in a large scale. In the CASHLP, both hub and nodes are unknown parameters, which classifies it to the typical NP. As one of the effective algorithms to solve NP, ACO has more and more research application in CSAHLP. However, ACO has its own weakness, too. With the conditions restricted by the capacity as the external constraints, ACO can not produce the correct solution sometimes. It gives a detailed analysis of the reasons for producing this infeasible solution and proper solutions to this problem and proves that the method is effective by experiment.

Key words: ACO; CSAHLP; NP; infeasible solution

0 引言

在大规模物流运输中, 比如说邮件配送、火车站物流等, 通常只是把这些物流通过一些中转点从始发地运送到目的地, 而这些始发地、目的地组成了带有物流权值的有向图。经过长期实践表明在这个有向图中把一些节点设置为中心点可以有效降低整体运输费用。中心点, 以后称之为枢流点, 是一些节点演化成具有中转功能的点^[1]。而轴-辐式网络在实际应用中, 其枢纽点的容量往往不可能无限制, 因此引入带有容量限制的轴辅式网络更有实际意义。文中将研究枢纽容量受限制的单分配轴-辐式网络选址问题 (capacitated single allocation hub location problems, 后文简称为

CSAHLP) 的解决方法及其修正因子^[2]。由意大利学者 M. DoRigo、V. Maniezz 和 A. Colomi 于 20 世纪 90 年代初提出的蚂蚁系统 (Ant System, 简称 AS) 优化而来^[3]。

蚁群算法已经成功应用在通讯、交通、人工智能等领域。相比于其他算法, 该算法具有明显的优越性。已经成功解决了车间调度问题 (Job Shop Scheduling, JSP), 二次指派问题 (Quadratic Assignment Problem, QAP) 等, 文中将详细讨论蚁群算法解决 CSAHLP 问题的方法和其中产生的非可行解处理方式。

1 枢纽容量受限的单分配轴-辐式网络设计模型

方程(1)表述了三次线性方程 (Linear Program) 描述 CSAHLP 问题的模型。在这里面枢纽点 p 是还没有确定了的, 同样这也是求解目标之一。

模型方程如下:

收稿日期: 2011-12-15; 修回日期: 2012-03-20

基金项目: 江苏省自然科学基金 (BK2008411)

作者简介: 胡 昊 (1985-), 男, 江苏扬州人, 硕士研究生, 研究方向为数据挖掘; 苏 勇, 副教授, 研究方向为知识发现与数据挖掘。

$$\min \sum_i \sum_k C_{ik} X_{ik} (\chi O_i + \delta D_i) + \sum_i \sum_k \sum_l \alpha C_{kl} Y_{kl}^i + \sum_k F_k X_{kk} \quad (1)$$

$$\text{s. t. } \sum_k X_{ik} = 1 \quad \forall i \in N \quad (2)$$

$$X_{ij} \leq X_{jj} \quad \forall i, j \in N \quad (3)$$

$$\sum_{i \in N} O_i X_{ik} \leq \Gamma_k X_{kk} \quad \forall k \in N \quad (4)$$

$$\sum_l Y_{kl}^i - \sum_l Y_{lk}^i = O_i X_{ik} - \sum_j W_{ij} X_{jk} \quad \forall i, k \in N \quad (5)$$

$$X_{ik} \in \{0, 1\} \quad \forall i, k \in N \quad (6)$$

$$Y_{kl}^i \geq 0 \quad \forall i, k, l \in N \quad (7)$$

方程目标函数为求解整个网络运输费用成本最低,此费用包括总的运输费用和枢纽节点设施固定费用之和^[4]。假设网络中有 n 个节点,用 $N = \{1, 2, \dots, n\}$, $i \in N$ 来表示这些节点的集合,其中的 p 个节点将被选做枢纽(hub)节点,其中的任意一对节点都可以作为交通流的起讫(OD)点。 W_{ij} 为从 i 点出发到 j 点的交通流, $O_i = \sum_j W_{ij}$, $D_i = \sum_j W_{ji}$ 。 C_{ij} 为从节点 i 至节点 j 的交通流的单位运输费用, α 为枢纽节点 k 与 l 之间的运输折扣系数,通常 $\alpha \leq 1.0$, χ 为非枢纽节点到枢纽节点的运输折扣系数, δ 为枢纽节点到非枢纽节点的折扣系数。假设在节点 j 处设置一个枢纽节点的固定费用为 F_j 。定义 Y_{kl}^i 为从 i 出发的所有经过枢纽节点 k 和 l 之间的交通流。 X_{ij} 为 $\{0, 1\}$ 整数变量,当节点 i 被分配给枢纽节点 X_{ij} 时为 1, 否则其值为 0, $X_{kk} = 1$ 则表示节点 k 为枢纽节点^[5]。其中 Γ_k 是枢纽 k 的容量。约束条件(2)为每一个非枢纽节点只能分配给一个枢纽节点的单分配约束,约束条件(3)确保了非枢纽节点只能分配给枢纽节点。约束条件(4)为枢纽节点的容量约束,约束条件(5)是商品流 i 在节点 k 的流量平衡方程,在节点 k 处的需求和供给由分配 X_{ik} 来决定。

对于此模型方程,特别提出如下几点注意事项^[6]:

1) 一个节点或枢纽点可能有到自身的流量,即 $W_{ii} \geq 0$ 。需要指出的是自身的流量并不是直接在自身处理,而是经过与其相连的枢纽点重新送回节点本身。这一点在以往文献中并未明确指出,是经过已知实验结果进行验证得到的。

2) 本问题中,枢纽点本身有到其他节点的流量而且还有它的“管辖”域里的节点到其他节点或枢纽点的流量。 W_{iklj} 流经枢纽点 i, j 时受到 Γ_k 及 Γ_j 约束,这使得目标方程(1)的可行域受到约束,从而减小了可行解域的空间。对于不确定的枢纽点 k , 假设有 $i_1, i_2, i_3, \dots, i_r$ 个节点属于其“管辖”域里,则对于 k 必须保证

$$\sum_r W_{iklj} \leq \Gamma_k \quad (\text{同(4)表述})^{[7]}。$$

3) 对于物流 W_{iklj} , 除枢纽 k, j 外不再考虑其他枢纽点的限制或传递作用。此条规则的现实意义即两个枢纽点可以直接沟通、传递物资,不需要经过第三方中转^[8]。

4) 对于枢纽点容量约束,并不将其看作一个背包问题。考虑到背包问题也是一个算法领域里典型 NP 问题,运算单独节点的容量规划并没有很好的时效比。相反,在实际处理中采用了先来先服务原则(First Come First Service, 简称 FCFS),以随机化节点堆栈来解决此问题。具体步骤为^[9]:

(1) 初始化备选城市,初始化枢纽点容量 Γ_k 。

(2) 随机化寻找备选城市 Random($i_1, i_2, i_3, \dots, i_r$)。

(3) 检查 $\Gamma_k - W_{ijk} \geq 0$ 是否成立,如果成立,将 i 加入 k 的管辖域,并进行计算 $\Gamma_k = \Gamma_k - W_{ijk}$ 。否则转回第一步。

(4) 检查 $\Gamma_k > 0$ 是否成立。若成立转回第一步,否则退出循环。

2 优化蚁群算法

蚁群算法可以看作为一种基于解空间参数化概率分布模型(Parameterized Probabilistic Model)的搜索算法框架(Model-based search algorithms)^[10]在蚁群算法中,解空间参数化概率,模型的参数就是信息素,因而这种参数化概率分布模型就是信息素模型。在基于模型的搜索算法框架中,可行解通过在一个解空间参数化概率分布模型上的搜索产生,此模型的参数用以前产生的解来更新,使得在新模型上的搜索能够集中在高质量的解搜索空间内^[11]。

蚁群优化算法在解决一般问题时分为这样几步:

1) 每一轮循环中,蚂蚁群组每一只蚂蚁在解决问题时都边移动边构成自己的解,直到蚂蚁构成所有解。

2) 在每一只蚂蚁完成自身解时,对解的构成进行信息素(Pheromone)更新。信息素是蚂蚁用来交换各自解的工具。信息素的更新使得解空间更趋向于优化区域。信息素的更新决定了下一只蚂蚁的移动方向和移动方向的概率,及其以后蚂蚁对于移动路径的选择。

更新所使用的公式如下:

$$\tau_{gh}(i, j) \leftarrow (1 - \rho) \cdot \tau_{gh}(i, j) + \rho \cdot \tau_{g0} \quad (8)$$

$\rho \in (0, 1)$ 为常数,是局部信息素挥发系数。

3) 每一只蚂蚁在移动过程中都会遇到路径的选择问题,它们使用伪随机比例选择规则(Pseudo random proportional action choice rule),并遵循这样的—个概率转移公式:

$$s = \begin{cases} \arg \max_{(i,j) \in J_{gh}(r)} [\tau_{gh}(i,j)]^\alpha [\eta_g(i,j)]^\beta, & \text{if } q \leq q_0 \\ (10) & \text{otherwise} \end{cases}$$

$$p_{gh}(s) = \begin{cases} \frac{[\tau_{gh}(i,j)]^\alpha \cdot [\eta_g(i,j)]^\beta}{\sum_{(i,j) \in J_{gh}(r)} [\tau_{gh}(i,j)]^\alpha \cdot [\eta_g(i,j)]^\beta}, & \text{if } s \in J_{gh}(r) \\ 0, & \text{otherwise} \end{cases} \quad (10)$$

其中 $q_0 \in (0,1)$ 为常数, $q \in (0,1)$ 为随机数, $\tau_{gh}(i,j)$ 表示解构成部分 (i,j) 的信息素, $\eta_g(i,j)$ 表示解构成 (i,j) 启发式因子, α 反应蚂蚁在运动中所积累的信息量(即残留信息量)在蚁群搜索中的相对重要程度, β 表示启发式因子的相对强弱。 $J_{gh}(r)$ 是那些被第 g 个群组中的第 h 个蚂蚁在步骤 r 中仍能选择加入到解中的部分。在选择下一解构成部分之前随机生成 q 。如果 q 的值小于等于常数 q_0 , 则从剩余所有可行的解构成中找出 $[\tau_{gh}(i,j)]^\alpha \cdot [\eta_g(i,j)]^\beta$ 最大的解构成, 即为下一个要选择的解构成; 如果 q 的值大于等于常数 q_0 , 则按(10)式中计算的概率来选择下一解构成。

4) 每一轮循环中, 目标函数值最优的那一只蚂蚁进行其解构成的全局信息素更新, 其使用全局更新公式:

$$\tau_g(i,j) \leftarrow (1 - \gamma) \cdot \tau_g(i,j) + \gamma \cdot \Delta\tau_g(i,j)$$

$$\Delta\tau_g(i,j) = \begin{cases} L_g^{-1}, & \text{if } (i,j) \in \text{the - best - solution} \\ 0, & \text{otherwise} \end{cases} \quad (11)$$

3 优化蚁群算法解决 CSAHLP

蚁群算法可以比较完整地解决此问题。结合禁忌表搜索方法(Tabu List Search)^[12], 这种算法显著收缩了每一步解空间的分布, 加快了收敛, 在有限次循环中得到较优的解。

在蚁群算法解决此问题中很重要的一个待定因素是启发式因子 η 的确定。 η 作为启发式因子有着非常重要的作用, 它必须反映出其他各个因素对于枢纽点分布的影响。 η 的设计需要考虑诸如总物流传输量的影响, 建造一个枢流点的整个成本影响, 还有自身流 W_{ii} 造成的目标函数值的影响。

总物流传输量 M —— M 是计算存储当前枢纽点分配方式产生的总体物流传输量。当解的结构不一样时, 整个的物流传输量也发生改变。虽然在总的物流不变的情况下, 不同的枢流分配方式使物流经过的枢纽点也不同, M 在蚁群运算的过程中成为衡量因素之

一。设计 M 的计算式为

$$M_i = \sum_{j=1}^N \sum_{l=1}^N W_{jl} (\chi d_{ji} + \delta d_{il}), 1 \leq i \leq N$$

建造费用 F —— F 的设计有很强的现实意义, 这一点体现在将一个节点转变为枢流点所需的费用上。建造费用 F 在蚁群算法的编程中, 以一个独立文件的形式提供, 每一个点都有其对应的费用, 此费用为当此点为枢纽点时, 所产生的额外费用。

自身流 Λ —— Λ 的设计是考虑当一个点对其本身有物流流量时的总体影响。设计 Λ 为: $\Lambda_i = \sum_{j=1}^s d_{xkji}$

综上: 设计

$$\eta_i = \frac{1}{K_i F_i \Lambda_i}, 1 \leq i \leq N \quad (12)$$

启发式因子 η 在整个运算中对于解的影响很大, 同样把它归结为修正因子。特别是它的出现修正了其他因子对于解的整体构成, 使得各个影响因素在解的函数值中有了具体的综合的影响。

优化蚁群算法解决 CSAHLP 问题步骤如下:

1) 确定每一轮蚂蚁数量 g (一般来说 $1 < g \leq \sqrt{1.44n}$, n 为城市的数量)。

2) 最大循环次数 $iItcount$ (为一般性约束条件, 在蚁群陷入过早的局部最优解时能约束在有限次运算)。对 n 个城市生成随机化序列, 即对 n 个城市进行随机排序, 对每个城市编号 $k_1, k_2, k_3, k_4, \dots, k_n$, 随机后形成 $g * iItcount$ 个 $k_{j_1}, k_{j_2}, k_{j_3}, \dots, k_{j_n} (j \in \text{random}(n))$ 序列, 保证每一只蚂蚁都将尽量获取不相同的排序序列, 这样做保证了每只蚂蚁解的多样性。

3) 初始化 CSAHLP 图。提前计算两个城市间距离 $distance(k_i, k_j)$, 初始化两个城市间的信息素 $m_dTrial(k_i, k_j)$, 根据 M. DoIrigio 的 ACS 理论, 任意两点之间初始化信息素为一个不为零的很小的信息素, 这里设置为 0.01, 这样做的目的是方便于信息素局部更新公式运算。初始化每个城市容量 $Cap(k_i)$ 。初始化每个城市需要运送的物流量 $Weight(k_i)$ 。

下面开始寻找解, 每一只蚂蚁在分布寻找路径时提前建立随机化序列栈 H , 栈的组成因子是城市序号。 H_1, H_2, H_3 分别对应随机选择城市 1, 随机选择城市 2, 随机选择城市 3。蚂蚁在每个城市 H_i 上, 都运用伪随机比例规则运算, 使用(9), (10)寻找其关系最强的城市, 经过多次迭代, 每个城市都找到与其联系最强的城市 j 后, 以 j 为 hub 建立其管辖域, 域成员为每个选择 j 为枢纽的城市 i ^[13]。

(1) 计算目标函数值。每一只蚂蚁走完途中所有点后, 对解对进行整理, 依据

$$L = \sum_i \sum_k C_{ik} X_{ik} (\chi O_i + \delta D_i) + \sum_i \sum_k \sum_l \alpha C_{kl} Y_{kl} +$$

$$\sum_k F_k X_{kk}$$

计算出目标函数值 $L^{[14]}$ 。

(2)局部信息素更新。每一只蚂蚁计算完目标函数值 L 后对自己的解依据(8)进行局部信息素更新。

(3)全局信息素更新。每一轮 $iltcount$ 中目标函数值 L 最小的那只蚂蚁再次根据自己的解对进行全局信息素更新。

(4)在有限次 $iltcount$ 中计算出较优解,保存。

在此过程中,特别强调“死蚂蚁”的处理。“死蚂蚁”是由前面提到过的对于容量处理时的 FCFS 原则影响下产生的,对于枢纽点 $k_1, k_2, k_3, \dots, k_p$ 其每个枢纽点潜在节点为 $i_1, i_2, i_3, \dots, i_r$ 个城市。因为使用了 FCFS 节点,较靠后的 i_{r-1}, i_r 点有较大的概率不能加入任何 $k_1, k_2, k_3, \dots, k_p$ 枢纽,且其联系其他枢纽点的测试概率 p 又较小,不足以作为单独的枢纽。这时,称寻找到这只解的蚂蚁为“死蚂蚁”。为解决此问题,在计算过程中设置了修正信号因子 V ,其值如下:

$$V = 1 \text{ or else}$$

在寻找解的过程中,若发现 V 为 1,则立刻停止蚂蚁的继续寻找,舍弃当前解,继续寻找下一组较优解。

4 实验数据

死蚂蚁并不是在每轮循环中都会产生。相比常解来讲出现的概率较小。在多次试验中每次出现的次数并不固定,也不稳定在某一个值域范围内。

以 20TT 为例,见表 1。

表 1 20TT 实验结果

第一轮	4
第二轮	2
第三轮	9
第四轮	1
第五轮	0
第六轮	8
第七轮	3
第八轮	9

以澳大利亚邮政数据为例,计算结果见表 2。

5 结束语

实验证实蚁群算法的确可以很好地解决 CSAHLP 问题,但其固有的缺点,导致其较快地收敛于较优解。

通过修改模型方程的线性约束条件、约束因子 W_{ii} , 和简化容量问题处理方式,降低了模型复杂度。将问题可行解约束在较小的一个范围内,这使得模型可以较好的被启发式算法所接受。

对于蚁群算法本身,在此问题中启发式因子起较

大的作用,好的启发式因子是决定性因素的集合的体现,这里 η 的设计极为巧妙,它综合考虑了总物流传输量、建造费用、自身物流这些影响因素,体现了启发式算法的数学基础。

表 2 澳大利亚邮政数据实验结果分析表

问题规模	目标函数 最优值	并行蚁群算法	并行蚁群算法
		平均偏差(%)	运算时间(s)
10LL	224250.05	2.17	0.06
10TL	263399.94	3.02	0.07
10LT	250992.26	1.54	0.08
10TT	263399.94	1.56	0.07
20LL	234690.96	0.70	0.42
20TL	271128.18	8.54	0.44
20LT	253517.4	3.59	0.48
20TT	296035.4	5.63	0.53
25LL	238977.95	7.92	0.83
25TL	310317.64	12.98	0.82
25LT	276372.5	4.12	0.86
25TT	348369.15	6.82	0.91
40LL	241955.71	13.36	2.99
40TL	298919.01	18.77	3.29
40LT	272218.32	9.34	3.09
40TT	354874.1	19.26	3.48
50LL	238520.59	18.28	5.86
50TL	319015.77	16.11	6.18
50LT	272897.49	14.11	6.15
50TT	417440.99	19.52	6.75

综合以上因素,修正因子的使用的确大大改善了结果和算式的数理体现,是必须和必要的。

参考文献:

[1] 徐晓华,陈 峻.一种自适应的蚂蚁聚类算法[J].软件学报,2006,17(9):1554-1559.

[2] 张 军,刘 羽,卢奉良.蚁群算法解决 TSP 问题的并行化研究与实现[J].计算机技术与发展,2011,21(5):72-74.

[3] 陈京民.数据仓库与数据挖掘技术[M].北京:电子工业出版社,2007.

[4] 张世永,钟亦平.基于蚁群算法实现入侵检测对动态网络拓扑的适应性[J].小型微型计算机系统,2005,26(7):1167-1173.

[5] 李士勇.蚁群优化算法及其应用研究进展[J].计算机测量与控制,2003,11(12):911-913.

[6] 吴 斌,赵燕伟.蚁群算法的研究现状[J].自动化仪表,2004,25(1):116-118.

[7] 毛嘉莉.聚类 K-means 算法及并行化研究[D].重庆:重庆大学,2003.

[8] 曾黄麟.粗集理论及其应用[M].重庆:重庆大学出版社,1996.

[9] Colormi A,Dorigo M,Maniezzo V. Distributed optimization by

(下转第 126 页)

节点数的增加吞吐率也在增加,当节点数达到一定个数时,吞吐率随着节点数的增加反而下降,MPQE 协议的吞吐率略高于其他协议。产生这种现象的原因是当节点数过少,资源利用不足;当节点数过多时,网络内的冲突现象占用过多的带宽等资源,导致分组丢失严重,从而吞吐率下降。

4.2 端到端平均时延

端到端的时延主要由路由建立所需的时间长度决定。从图 3 可以看出,随着节点数的增加,几种协议的端到端时延逐渐增加。MPQE 协议的延时要低于其他三种协议,且 DSR 协议随着节点数的增加时延增加幅度最大。产生这种现象的原因是随着节点数的增加,网络拥塞状况不断恶化,由于 MPQE 协议考虑了节点的可用带宽和剩余能量,因此能够采用相对稳定的路径进行数据传送,从而端到端时延也得到了很好的改善。

4.3 网络生命周期

网络生命周期是网络在运行过程中,第一个节点死亡的时间。从图 4 可以看出,随着节点数的增加,几种协议的网络生命周期逐渐降低。MPQE 协议的网络生命周期要高于其他三种协议,且 DSR 协议随着节点数的增加生命周期下降幅度最大。这一现象产生的原因是 MPQE 协议均衡了节点可用带宽和网络能量消耗,减少能量低的节点参与路由的机会。因此,延长了网络的生命周期。

5 结束语

为了提供 Ad Hoc 网络的端到端 QoS 支持,文中提出了基于能量约束的多径 QoS 路由协议(MPQE)。该协议利用节点的可用带宽和剩余能量的大小进行路由寻路,提高网络的吞吐率并延长了网络的生命周期,此外,多径路由的提供减少了路由重新发现的频率并提高了数据的传输效率。

参考文献:

- [1] 于宏毅. 无线移动自组织网络[M]. 北京:人民邮电出版社,2005:171-176.
- [2] Perkins C E, Royer E M. Ad-hoc on-demand distance vector routing[C]//The Second IEEE Workshop on Mobile Computing System and Applications. [s. l.]: [s. n.], 1999: 90-100.
- [3] Zhong Yingji, Yuan Dongfeng. Dynamic Source Routing Protocol for Wireless Ad Hoc Networks in Special Scenario Using Location Information[C]//International Conference on Communication Technology. [s. l.]: [s. n.], 2003:1287-1290.
- [4] 陈蓉,李喆. AD HOC 网络中的一种 QoS-AWARE 多径路由协议[J]. 系统仿真学报,2009(3):1390-1394.
- [5] 余萍. 一种基于可用带宽的 QoS 多播路由算法[J]. 计算机工程与设计,2006(8):2976-2978.
- [6] Sun Baolin, Gui Chao, Liu Pengyuan. Energy Entropy Multipath Routing Optimization Algorithm in MANET Based on GA[C]//The Fifth IEEE International Conference on Bio-inspired Computing: Theories and Applications (BIC-TA). [s. l.]: [s. n.], 2010:943-947.
- [7] Leung R, Liu Jilei, Poon E, et al. MP-DSR: a QoS-aware multi-path dynamic source routing protocol for wireless ad-hoc networks[C]//The 26th Annual IEEE Conference on Local Computer Networks. [s. l.]: [s. n.], 2001:132-141.
- [8] 成小惠. 一种基于能量感知的节点独立多径路由协议[J]. 中国电子科学研究院学报,2010(4):173-178.
- [9] Hamrahi Z, Ghaffari A, Rasi T. MPRGQ: multi-path routing algorithm to guarantee to achieve the required QoS of wireless sensor networks[C]//2010 International Conference on Signal and Image Processing (ICSIP). [s. l.]: [s. n.], 2010:430-435.
- [10] 张国胜,罗菊. 一种基于路径稳定性的 MAODV 路由协议改进算法[J]. 数字通信,2010(1):77-79.
- [11] 柯志亨,程荣祥,邓德隽. NS2 仿真实验-多媒体和无线网络通信[M]. 北京:电子工业出版社,2009:1-23.
- [12] 蒋晶晶. 基于 NS2 的 Ad Hoc 网络路由仿真研究[D]. 武汉:华中师范大学,2009.

(上接第 122 页)

- ant colonies[C]//First Eur. Conference on Artificial Life. [s. l.]: [s. n.], 1991:134-142.
- [10] Dorigo M, Maniezzo V, Colomi A. 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] Dorigo J M, Gambardella L M. Ant colony system: a cooperative learning approach to the traveling salesman problem[J]. IEEE Transactions on Evolutionary Computation, 1997, 1(1):

53-66.

- [12] Bland J A. Space-planning by ant colony optimization[J]. International Journal of Computer Applications in Technology, 1999, 12(6):320-328.
- [13] Dorigo M, Caro G D, Gambardella L M. Ant algorithms for discrete optimization[J]. Artificial Life, 1999, 5(2):137-172.
- [14] Bullnheimer B, Kotsis G, Strauss C. Parallelization strategies for the ant system[R]. Vienna: University of Vienna, 1997.

基于改进 BP 网络模型的公路流量预测

作者:

作者单位:

刊名:

英文刊名:

年, 卷(期):

彭勇, 陈俞强, 严文杰

彭勇,陈俞强(东莞职业技术学院计算机工程系,广东东莞523800), 严文杰(武汉理工大学计算机科学与技术学院,湖北武汉430070)

计算机技术与发展

Computer Technology and Development

2012(8)

参考文献(14条)

- 徐晓华;陈斌 一种自适应的蚂蚁聚类算法[期刊论文]-软件学报 2006(09)
- 张军;刘羽;卢奉良 蚁群算法解决TSP问题的并行化研究与实现[期刊论文]-计算机技术与发展 2011(05)
- 陈京民 数据仓库与数据挖掘技术 2007
- 张世永;钟亦平 基于蚁群算法实现入侵检测对动态网络拓扑的适应性[期刊论文]-小型微型计算机系统 2005(07)
- 李士勇 蚁群优化算法及其应用研究进展[期刊论文]-计算机测量与控制 2003(12)
- 吴斌;赵燕伟 蚁群算法的研究现状[期刊论文]-自动化仪表 2004(01)
- 毛嘉莉 聚类K-means算法及并行化研究[学位论文] 2003
- 曾黄麟 粗集理论及其应用 1996
- Colomi A;Dorigo M;Maniezzo V Distributed optimization by ant colonies 1991
- Dorigo M;Maniezzo V;Colomi A Ant system:optimization by a colony of cooperating agents 1996(01)
- Dorigo J M;Gambardella L M Ant colony system:a cooperative learning approach to the traveling salesman problem 1997(01)
- Bland J A Space-planning by ant colony optimization[外文期刊] 1999(06)
- Dorigo M;Caro G D;Gambardella L M Ant algorithms for discrete optimization 1999(02)
- Bullnheimer B;Kotsis G;Strauss C Parallelization strategies for the ant system 1997

本文链接: http://d.g.wanfangdata.com.cn/Periodical_wjtz201208031.aspx