

基于改进伊藤算法的最短路径网络路由优化算法

满振祯 余世明 何德峰

(浙江工业大学信息工程学院 杭州 310023)

摘 要 通过对网络路由最短路径问题进行分析,使用伊藤算法求解以费用最低为目标的路由优化问题,建立最短路径路由问题的网络结构模型。为加快伊藤算法求解费用最低路由的收敛速度,在状态转移策略中引入费用启发因子,优化漂移和波动过程,并改进路径权重更新规则。将种群交叉思想引入算法中,利用种群间的信息交流加快了算法的收敛速度并提高了寻优能力。在2-opt算子局部优化的基础上加入反转算子,避免陷入局部最优解。文中还对算法的收敛性进行了系统分析。实验结果表明,改进后的算法有效提升了收敛速度并加强了寻优能力。

关键词 路由优化,最短路径,伊藤算法,网络优化

中图法分类号 TP393

文献标识码 A

DOI 10.11896/j.issn.1002-137X.2017.07.038

Shortest Path Network Routing Optimization Algorithm Based on Improved ITO Algorithm

MAN Zhen-zhen YU Shi-ming HE De-feng

(College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China)

Abstract Through the analysis of shortest path problem on network routing, we built the network model of the shortest path routing problem by using ITO algorithm to get the lowest cost route optimization target. To speed up ITO lowest routing algorithm's convergence speed, the proposed algorithm involves cost inspired factor in the state transition strategy, and hence improves path weight updating rules. The convergence speed and optimization capability are enhanced simultaneously by introducing the method of population cross and using the information exchange between populations. A 2-opt operator-based inversion operator is adopted to optimize the algorithm and to avoid falling into local optimal solution. The astringency of the algorithm was also analyzed systematically. The case study illustrates that the improved algorithm is effective to speed up the convergence speed and enhance optimization capability.

Keywords Route optimization, Shortest path, ITO algorithms, Network optimization

1 引言

随着 Internet 的广泛应用和通信技术的飞速发展,人们的生活和工作也越来越离不开通信网络的应用。网络通信由于涉及到多用户参与,因此要消耗大量的网络资源^[1]。

路由优化的目标就是找到一种算法或策略,在给定网络和路由需求的情况下,寻求一种链路连接方式,使网络资源能够得到有效利用。最短路径不仅指一般意义上结点之间的距离最短,还可以引申到其他度量指标,如延时、费用、延时抖动等 QoS 约束。因此,最短路径问题也可以转变为最低费用等问题。各国学者经过探索研究,提出了一些快速有效的算法,如基于最短路径的 Dijkstra 算法,即计算源节点到目的节点的最短路径;求最小网络代价应用的斯坦利(Steiner)树路由算法使得在任一对源节点和目的节点之间都存在通路,并使其代价最小。文献[2]列出了一些比较典型的启发式算法,这些算法的缺点是运算开销过大,不能应用到大规模的网络中。

考虑到传统最短路径搜索算法,如宽度优先搜索算法、Dijkstra 算法、Bellman-Ford 算法等存在不足,人们开始探索将智能算法引入路由问题的求解中。文献[3-4]提出的基于遗传算法的路由方案较好地克服了启发式算法的相关缺点,获得了较好的效果;但由于自身的一些不足,如存在早熟现象,可能会使搜索陷入局部最优。蚁群算法和遗传算法都具有全局优化和本质上的并行性,文献[5]基于蚁群算法弥补了 Dijkstra 算法搜索时间过长和容易陷入局部最优解的不足,使运算速度明显加快。但是,在实时通信环境、高动态拓扑结构、QoS 要求的网络结构的情况下,传统的最短路径搜索算法的计算复杂度高、效率低、寻优时间长。

伊藤算法^[6-8](ITO Algorithm, ITO)是由董文永教授在各种启发式群智能算法相互融合的发展形势下提出的,与遗传算法、模拟退火算法等早期进化算法相比,具备鲁棒性更强、求解时间更短、易于计算机实现等优点,在求解组合优化、函数优化、系统辨识、时间序列建模等问题时取得了很好的效

到稿日期:2016-06-05 返修日期:2016-09-14

满振祯(1989—),男,硕士生,主要研究方向为智能启发式算法及嵌入式系统;余世明(1962—),男,博士,教授,博士生导师,主要研究方向为模型预测控制、大系统建模与优化、数字控制理论及其在机电一体化系统中的应用,E-mail:ysm@zjut.edu.cn;何德峰(1979—),男,博士,教授,硕士生导师,主要研究方向为模型预测控制及优化计算。

果。但从目前刊出的文献来看,伊藤算法的研究应用还未涉及路由算法优化领域。文献[9]在理论上深度分析了伊藤算法的收敛性及期望收敛速度,并且以图模型为基础对算法中的各算子参数进行了研究。文献[10]以基本伊藤算法为框架,通过融入蚁群算法中信息素的设计重新构造了粒子状态转移策略,并将其用于求解基于交通因素、客户需求动态改变、用户满意度的多目标动态车辆路径问题。易云飞等^[11]为了提高伊藤算法求解离散组合优化问题的效率,引入了协同扩散过程的漂移系数,采用局部搜索能力强的爬山法来确定波动系数,并将改进后的伊藤算法用于求解带软时间窗的车辆路径问题。在求解 VRP 问题的过程中,针对选取客户节点时易陷入局部最优和迭代陷入停滞的现象,文献[12]引入融合路径权重和距离启发因子的节约法决策规则。目前,也有学者开始将伊藤算法的优点与其他算法相结合,以提高算法的寻优能力。文献[13]利用伊藤算法漂移和波动的特点改进人工蜂群算法采蜜蜂和观察蜂的搜索方式,改善了人工蜂群算法求解复杂问题时出现的收敛速度慢、易陷入局部最优的问题。

本文采用新型启发式伊藤算法研究以最低费用为目标的路由优化问题,该算法已在求解 TSP 中取得了较好的实验结果。TSP 问题就是在给定的多个城市中寻找一条长度最短的闭合路径,此路径有且仅有一次历经每个城市,最后返回始发城市。最短路径路由优化问题类似于 TSP 问题。针对基本伊藤算法,改进路径权重更新策略;同时运用种群交叉进行信息交流,避免信息浪费,加快了算法的收敛速度并提升了寻找最优路径的能力;最后利用局部 2-opt 和反转优化算子进一步完成全局寻优。仿真测试证明了所提 ITO 算法是有效、可行的。

2 网络模型

研究路由问题时,可以将网络用一个无向带权连通图 $G=(V,E)$ 表示,其中 V 表示节点集, E 表示连接节点的链路集,设相邻两个节点之间最多仅有一条可通信链路。 $|V|$ 和 $|E|$ 分别表示网络中的节点数和链路数, C_{ij} 为链路 (i,j) 的费用,费用矩阵为 $C=[C_{ij}]$ 。源节点和目的节点分别用 S 和 D 表示,每条链路的连接用 I_{ij} 表示,定义如下:

$$I_{ij} = \begin{cases} 1, & \text{节点 } i \text{ 与节点 } j \text{ 之间存在路径} \\ 0, & \text{节点 } i \text{ 与节点 } j \text{ 之间不存在路径} \end{cases}$$

由上式易知, I_{ij} 的对角线元素为 0,同时满足条件:若 $I_{ij}=1$,且 $I_{jk}=1$,则 $I_{ik}=1$ 。使用以上定义,最短路径问题可描述为最小值的优化问题,目标函数表示为:

$$\min \sum_{i=S}^D \sum_{j=S}^D C_{ij} \cdot I_{ij} \quad (1)$$

$$\text{s. t. } \sum_{j=S}^D I_{ij} - \sum_{j=S}^D I_{ji} = \begin{cases} 1, & \text{if } i=S \\ -1, & \text{if } i=D \\ 0, & \text{otherwise} \end{cases}$$

$$\sum_{j=S}^D I_{ij} \leq 1, \quad \text{if } i \neq D \\ \sum_{j=S}^D I_{ij} = 0, \quad \text{if } i=D$$

$$I_{ij} \in \{0,1\}, \text{ for all } i \quad (2)$$

式(2)保证了源节点与目的节点之间的路径最短。

3 求解路由问题的基本伊藤算法

伊藤算法是基于伊藤过程的智能算法,将粒子系统所在空间映射为优化问题的搜索空间,在求解路由优化问题时,将数据包访问节点的过程映射为粒子运动的过程。每个粒子根据初始路径权重,按照轮盘赌的方式漂移和波动至下一个节点,当粒子从源节点到达目的节点后,计算粒子走过的所有可达路径的费用,并找出最优粒子以更新全局最优解。同时,更新所有粒子的半径、漂移算子和波动算子的大小,模拟退火之后,进入下一次迭代,直到算法满足迭代终止条件。

3.1 粒子漂移和波动策略

伊藤算法受粒子热运动启示,粒子运动由漂移和波动过程组成,漂移过程体现算法的向优性,在宏观上决定粒子向吸引元方向(最优路径)移动;波动过程体现算法的探索能力,粒子在邻域内随机进行波动,从而达到精细化搜索和跳出局部最优解。粒子向一个节点转移的过程是先按照漂移算子设定的路径权重进行漂移,按式(3)求得向每个可选节点的转移概率,再采用轮盘赌策略选择完成。漂移完成后,按波动算子更新路径权重,采用同样的方式进行波动,以完成节点间的状态转移。

$$p^m(i,j) = \begin{cases} \frac{\tau(i,j)}{\sum_{l \in \text{tabu}_m} \tau(i,l)}, & i \in \text{tabu}_m, j \notin \text{tabu}_m \\ 0, & \text{else} \end{cases} \quad (3)$$

其中, $p^m(i,j)$ 表示粒子 m 从节点 i 向下一节点 j 转移的概率, tabu_m 用于存储粒子 m 已访问过节点的禁忌表, $\tau(i,j)$ 表示算法中由漂移算子或波动算子设定的节点之间的路径权重, l 表示待访问的节点。其中路径权重 $\tau(i,j)$ 在漂移过程按照式(4)所求得的漂移路径权重 $\tau_\mu(i,j)$ 进行更新;粒子完成漂移后,按照式(5)进行波动过程中的波动路径权重 $\tau_\rho(i,j)$ 更新。

$$\tau_\mu(i,j) = \begin{cases} 1, & \text{if } e(i,j) \in \sigma \cap e(i,j) \in \sigma' \\ \mu, & \text{if } e(i,j) \in \sigma' \\ 1-\mu, & \text{if } e(i,j) \in \sigma \\ 0, & \text{else} \end{cases} \quad (4)$$

$$\tau_\rho(i,j) = \begin{cases} 1-\rho, & \text{if } e(i,j) \in \sigma \\ \frac{\rho}{n-1}, & \text{else} \end{cases} \quad (5)$$

其中, $e(i,j)$ 表示两节点 i 和 j 之间的路径段, σ 表示当前粒子上一次迭代的路径, σ' 表示最优路径, μ 和 ρ 分别表示漂移算子和波动算子强度, n 表示波动可选节点数。

3.2 漂移算子和波动算子的设计

波动算子主要完成粒子在自己邻域内的局部扰动,漂移算子主要实现粒子朝着吸引元的方向前进。伊藤算法受粒子热运动启示,漂移和波动的剧烈程度由环境温度和粒子半径共同决定。因此,为简化起见,通过将两个因子映射到算法中求取漂移和波动算子的函数表达式为:

$$\rho = \mu = \omega_{\min} + f_1(r) \cdot f_2(t) \cdot (\omega_{\max} - \omega_{\min}) \quad (6)$$

其中, $f_1(r) = \frac{e^{-\lambda r} - e^{-\lambda r_{\max}}}{e^{-\lambda r_{\min}} - e^{-\lambda r_{\max}}}$, $f_2(t) = e^{-\frac{1}{t}}$ 分别表示粒子半径和环境温度对漂移和波动算子强度的影响, λ 为常数, r 为粒子半径, t 表示环境温度。 $\omega_{\max}$ 和 $\omega_{\min}$ 分别表示最大和最小

波动(漂移)强度,一般情况取 $\omega_{\max} = 1 - \omega_{\min}$, 本文取 $\omega_{\max} = 1$, $\omega_{\min} = 0$ 。

3.3 粒子半径的设计

将半径的设计与布朗运动中的运动规律结合起来,根据爱因斯坦和郎之万的公式,半径越小时粒子运动越剧烈,映射到算法中的漂移和波动算子的值也就越大,因此它就会大范围地运动,以达到快速搜索的目的,这更利于大范围搜索全局最优解。同时,粒子半径也不是恒定不变的,它根据当前适度值(即目标函数值)动态变化,继而影响漂移和波动的强度,这也是伊藤算法自适应的特点。由于本文的目标函数为极小化问题,粒子半径函数 $r(x) = g(f(x))$, g 应为单调减函数,采用基于排序的方法对种群中的 M 个粒子按照目标函数值由小到大进行排序 $\{L_1, L_2, \dots, L_n\}$, 并按式(7)求取半径。

$$r_{L_i} = r_{\max} - \frac{r_{\max} - r_{\min}}{M-1} \times (L_i - 1) \quad (7)$$

其中, M 表示粒子个数, L_i 表示粒子排列号, 通常情况取 $r_{\max} = 1$, $r_{\min} = 0$ 。

3.4 环境温度的设计

伊藤算法以粒子热运动为基础,算法中环境温度的设计也遵从布朗运动的运动规律,环境温度越高,粒子运动越剧烈,映射到算法中的漂移和波动过程也越剧烈。伊藤算法的温度变化过程采用模拟退火算法中的设计方法,通过冷却进度表控制环境温度,经过一定次数的迭代退火后,控制算法收敛。退火过程如式(8)所示:

$$t = t_k \times \text{des_speed} \quad (8)$$

其中, k 表示温度下降次数($k=0$ 为初始温度), des_speed 表示退火速度($\text{des_speed} \in (0, 1)$)。

4 求解路由优化问题的改进伊藤算法

4.1 改进的路径权重更新规则

由基本伊藤算法的式(4)和式(5)发现,在传统的伊藤算法中,漂移过程和波动过程单独进行,首先粒子根据波动策略更新边权重,并根据更新后的边权重更新当前解,完成波动过程,然后粒子根据漂移策略更新边权重,再更新当前解,完成漂移过程,导致算法的复杂度增加。但是,粒子在向吸引元方向实际运动的过程中,漂移和波动是同步进行的,文献[14]在求解 VRP 问题时,提出结合漂移和波动过程更新路径权重的方法。因此,本文决定根据在粒子搜索过程中目标函数值逐渐优化以及粒子半径、漂移和波动算子自适应改变的特性,综合来考虑改进路径权重更新规则,使漂移和波动不再单独进行,更符合实际粒子运动规律特性,并且有效降低了算法的复杂度。优化的路径权重更新规则如下:

$$\tau(i, j) = \begin{cases} 2 - \rho, & \text{if } e(i, j) \in \sigma \cap e(i, j) \in \sigma' \\ 1 + \rho + \mu, & \text{if } e(i, j) \in \sigma' \\ 2 - \rho - \mu, & \text{if } e(i, j) \in \sigma \\ \rho + \mu, & \text{else} \end{cases} \quad (9)$$

其中, σ 表示当前粒子上一次迭代的路径, σ' 表示当前最优路径, μ 和 ρ 分别表示漂移算子和波动算子的强度, $e(i, j)$ 表示两节点 i 和 j 之间的路径段。

4.2 改进的路径选择策略

基本伊藤算法中粒子根据路径权重选择节点,但是初始

时刻路径权重全部相同,搜索具有盲目性,算法收敛速度较慢,在迭代初始阶段粒子的运动一般会对最终结果产生影响,因为后续路径权重的更新是根据上一次迭代的结果自适应改变的。在求解 TSP 和 VRP 问题时,有文献提出距离启发因子或者节约里程表的方法,但是在数据包传输过程中,本文旨在寻找最低费用的最优路径,因此,距离启发因子已不具有实际意义;另外,对算法结果好坏的评判依据是粒子所经过路径的总费用,式(3)的选择策略没有体现出费用因素。本文的状态转移策略从全局优化角度出发,考虑并引入费用启发因子 $\eta(i, j) = 1/C_{ij}$, 通过结合路径权重和费用启发因子设计路径选择策略:

$$p^m(i, j) = \begin{cases} \frac{[\tau(i, j)]^\alpha [\eta(i, j)]^\beta}{\sum_{l \notin \text{tabu}_m} [\tau(i, l)]^\alpha [\eta(i, l)]^\beta}, & i \in \text{tabu}_m, j \notin \text{tabu}_m \\ 0, & \text{else} \end{cases} \quad (10)$$

其中, tabu_m 用于存储粒子 m 已访问过节点的禁忌表, $\tau(i, j)$ 表示算法中由漂移算子或波动算子设定的节点之间的路径权重, α 表示路径权重对节点选择的影响因子, β 表示费用启发因子对节点选择的影响因子, l 表示待访问的节点。

4.3 种群交叉伊藤算法

在当前各种种群智能算法相互融合的发展形势下,本文将种群交叉思想引入伊藤算法中用于求解路由费用最低的路径优化问题。

为了方便算法处理,一般假设粒子半径均匀分布在 $[r_{\min}, r_{\max}]$ 之间,这样可以使得不同粒子的搜索性能不同,保证粒子的多样性,加大算法的搜索力度。同时,为了防止超级粒子的出现,种群中的粒子半径不能有太大差别。

首先,粒子在初始种群中进行目标搜索,算法进行数代后会陷入局部最优解进而出现搜索停滞现象。当算法运行了一定代数但最优解未更新时,便按照适度值的大小将粒子均分为两个新的子种群,将适度值较好的粒子分配到第一子种群,将适度值较差的粒子分配到第二子种群,再分别对两个子种群进行目标搜索。

第一子种群的适度值较好,粒子半径大,运动幅度小,不利于算法跳出局部最优解,随着搜索的继续进行,群体多样性迅速减少;此时第二子种群的适度值较差,半径较小,运动范围较大,将此子种群的粒子半径依次分配给适度值较好的第一子种群,以增强第一子种群粒子漂移和波动算子的强度,使得第一子种群的粒子在搜索空间进行更广泛、快速的搜索而跳出局部最优解,进而增加解的多样性。

对于适度值较差的第二子种群,仍按照自己的粒子半径进行目标搜寻。当搜索进行了一定的代数,最优解有所改进且未满足终止条件时,将两个子种群再交叉混合成一个种群,继续按照初始设计的漂移和波动算子计算公式进行搜索,以寻找最优解。

4.4 局部优化算法

为进一步改善算法的收敛速度和全局最优解,本文针对每代最优解引入 2-opt 优化算子和反转算子。2-opt 以遍历的方式在路径内部选中的位置 i 和 j 之间进行两两元素的互换调整;反转算子在随机选中 i 到 j 之间的元素反转后,以逆

序的方式重新插入。局部搜索不能仅局限于子路径内部的调整,各子路径之间的调整能更好地增大寻优空间,并且邻域搜索种类越丰富,在当前最优解邻域内搜索到更优解的概率就越大。

为简要说明局部优化调整,假设算法持续一定代数后已经获得路径: $Trace1=\{S,3,4,9,13,14,D\}$, $Trace2=\{S,2,3,13,7,15,18,D\}$, $\dots$, $TraceN=\{S,\dots,D\}$;取 $Trace[i,j]$ ($i<j,i\neq S,D;j\neq S,D$) 为当前路径的子路径。例如, $Trace1[3,5]=\{4,9,13\}$, $Trace2[4,5]=\{13,7\}$ 为其对应当前路径的子路径。 $Trace1$ 经过 2-opt 算子的内部优化调整后, $Trace1^1=\{9,4,13\}$, $Trace1^2=\{13,9,4\}$, $Trace1^3=\{4,13,9\}$ 是新的子路径,重新插入。 $Trace1$ 经过局部调整后,产生 3 种新的路径。反转算子则把子路径 $Trace1[3,5]$ 作逆序调整,除重新插入产生新的路径外,也用于完成路径间的优化调整。例如, $Trace2$ 逆序后 $Trace2[4,5]=\{7,13\}$,可用于插入 $Trace1$ 以替换子路径 $Trace1[3,5]$,继而产生新的路径 $Trace1=\{S,3,7,13,14,D\}$ 。

每进行一次局部优化,便计算产生的新路径的目标函数值,并判断其是否优于当前全局最优值,若优于,则替换全局最优值。

4.5 算法收敛性分析

为了证明改进后的伊藤算法的全局收敛性,将伊藤算法的迭代过程映射为马尔科夫(Markov)链,然后采用 Markov 链分析算法的全局收敛性^[11]。

将伊藤随机过程定义如下:若有随机过程 $X=\{X(t),t\geq 0\}$ 满足以下伊藤积分等式: $\forall 0\leq t_0<t<T$,使

$$X(t)-X(t_0)=\int_{t_0}^t b(s,X(s))ds+\int_{t_0}^t \delta(s,X(s))dB(s)$$

(11)

或者满足以下伊藤微分方程等式:

$$dX(t)=b(t,X(t))dt+\delta(t,X(t))dB(t)$$

(12)

则 X 被称为伊藤随机过程。式(12)中, $b(t,X(T))$ 表示漂移率, $\delta(t,X(t))$ 表示干扰强度。

随机过程 $X_T=\{X_T,t\in T=(0,1,2,\dots)\}$,其状态空间为 $\Omega=\{0,1,2,\dots\}$,若对任意正整数 k ,任意 $t_i\in T,t_i<t_{i+1}$ ($i=$

$0,1,2,\dots,k+1$) 及任意非负整数 $i_0,i_1,i_2,\dots,i_{k+1}\in\Omega$,只要: $P(X_{t_0}=i_0,X_{t_1}=i_1,X_{t_k}=i_k)>0$,总有: $P(X_{t_{k+1}}=i_{k+1}|X_{t_0}=i_0,X_{t_1}=i_1,\dots,X_{t_k}=i_k)=P(X_{t_{k+1}}=i_{k+1}|X_{t_k}=i_k)$ 成立,则称 X_T 为离散时间的 Markov 链。

根据伊藤积分公式,首先可以把漂移过程理解为所有的解都应该朝着当前吸引元的方向移动,以提高自己的评估值;其次,伊藤方程中的漂移可理解为粒子的向优性,之后再对其进行波动。

定理 1 伊藤算法的运行过程是时间齐次的有限马尔科夫链。

其证明过程可参考文献[11]。

定理 2 求解最短路径路由优化问题的伊藤算法最终以概率 1 收敛到全局最优解。

证明:设 Ω^* 为包含全局最优解的所有子群体的集合。由 Markov 链理论可知, $(\Omega-\Omega^*)$ 中的状态是非常返状态,而 Ω^* 中的状态是常返状态。如果伊藤算法从某个非常返状态 $(\Omega-\Omega^*)$ 中的状态出发时粒子可能在非常返集合 N 中运动,则由文献[11]中的引理 1 可知,粒子以概率 1 必定到达常返状态。同时由引理 2^[11] 可知,粒子在某个时刻也可能进入到某个闭集中运动,之后就一直在该闭集中运动;如果从某一常返状态 Ω^* 中的状态出发,粒子就会永久在该常返状态所在的闭集中运动。因此,求解最短路径路由优化的伊藤算法中的粒子无论从非常返状态集合还是常返状态集合出发,最终都能以概率 1 收敛到全局最优解。

5 算法仿真实验与结果分析

5.1 网络结构的仿真实验

为对比验证所提算法的有效性,将其与 Munemoto 算法^[15]、Inagaki 算法^[16]、Ahn 算法^[17] 和孙宝林等提出的算法^[18] 进行比较,采用文献[18]的网络拓扑结构(即网络节点数为 20)进行验证。实验在 MatlabR2013b 上对每种算法求得的最优解进行多次仿真测试。实验具体参数的设置如表 1 所列,其中基本 ITO 算法无费用启发因子项,路径权重影响因子 $\alpha=1$ 。

表 1 实验参数设置

粒子个数 M	最大迭代次数 N_{max}	路径权重影响 因子 α	费用影响 因子 β	初始环境温度 T_0	退火速度 des_speed	每次温度迭代长度 T_Length	最优路径未 更新代数
50	10	5	3	8000	0.98	4	5

网络结构及算法结果如图 1—图 5 所示。

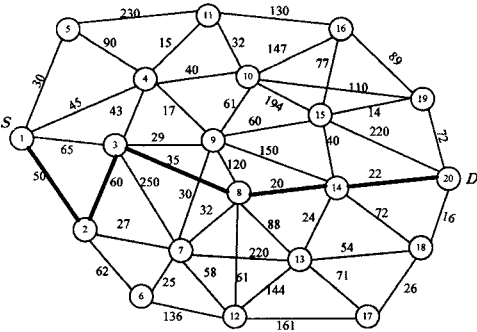


图 1 Munemoto 算法的计算结果(总费用 187)

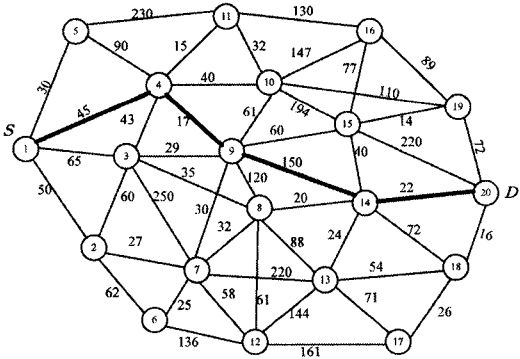


图 2 Inagaki 算法的计算结果(总费用 234)

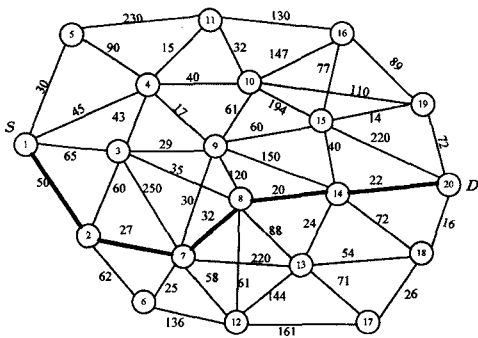


图3 基本 ITO 算法的计算结果(总费用 151)

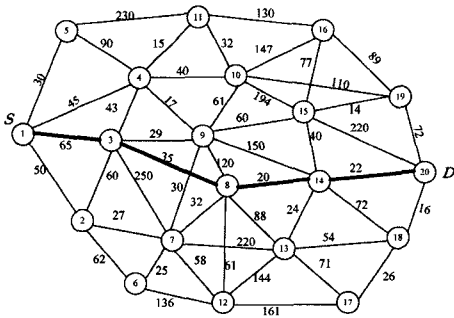


图4 改进 ITO 算法与 Ahn 算法、孙宝林算法的计算结果(总费用 142)

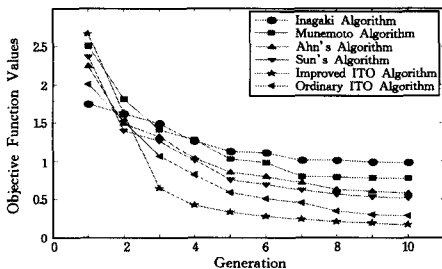


图5 算法收敛特性

5.2 随机生成网络拓扑结构的仿真实验

本小节采用 Salam 网络拓扑随机生成算法生成 15~50 个节点的对称网络结构,使用 K 均值聚类控制节点分布的疏密,使得产生的网络拓扑的连通性和均匀性更好;随机产生链路费用,以验证本文改进的 ITO 算法的通用性和有效性。仿真实验的现实意义是验证本文算法可扩展到较大的网络结构。

路由优化失败率定义为多次运行算法时不能找到全局最优路径的次数所占的百分比。可以认为路径失败的相对频率接近于所计算的路径不是全局最优路径的概率。仿真所用对比的遗传算法的种群数量设置与网络节点数相同。每种节点数情况下随机生成 20 种网络拓扑结构,对应每种网络拓扑结构各算法均运行 50 次节点数,失败率取平均值。

5.3 实验结果分析

图 1—图 4 所示为算法所求得的从源节点到目的节点的最低费用路径(粗实线表示),通过本文 ITO 算法与改进的 Ahn 遗传算法和文献[18]算法计算的最小费用一致。图 5 是本文算法与各算法的全局收敛性比较,从图中可以看出,改进的 ITO 算法在 Generation 迭代次数为 3 时就已经逼近文献

[18]的收敛性,表明本文的收敛速度较其他算法快,全局收敛性效果更好。图 6 示出不同节点数的情况下本文改进的 ITO 算法与其他遗传算法的优化失败率,从中看出改进 ITO 算法的计算优化失败率较其他算法低,算法表现较好,在不同网络结构下性能表现较稳定,寻优能力更强。以上说明,本文 ITO 算法的全局寻优及收敛性较好,收敛速度快于其他遗传算法,可靠性更高,能满足实时通信环境、高动态拓扑结构以及 QoS 单播或多播路由的网络结构需求。

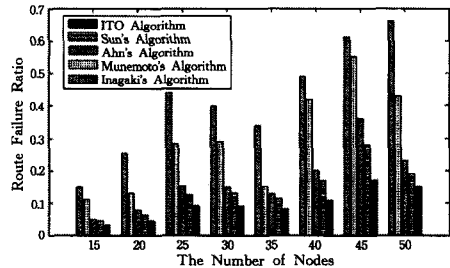


图6 各算法路由最短路径优化率的比较

结束语 针对路由优化问题,在现有文献的基础上通过分析最短路径问题,结合伊藤算法在迭代过程中的特性以及现有伊藤算法进行了改进,并将其用于求解路由优化问题。根据理论分析和多次实验选取了较优参数配置,通过对比现有遗传算法,表明改进后的伊藤算法有效加快了算法的收敛速度并提升了全局搜索最优解的能力。本文算法虽然在实验中取得了不错的效果,但仍然存在一些不足之处,今后的研究方向是进一步提升伊藤算法的性能,并将其应用于复杂的路由优化领域。

参考文献

- [1] ZOU E, LIU Z H, FANG S Y, et al. Research on Multicast Routing Optimization Based on Chaos Genetic Algorithm[J]. Computer Engineering, 2011, 37(3): 155-157. (in Chinese)
- [2] FLEISCHER M. The Measure of Pareto Optima: Applications to Multiobjective Metaheuristics[M]// Evolutionary Multiobjective Optimization. Berlin, Germany: Springer-Verlag, 2003: 519-533.
- [3] WHILE L, HINGSTON P, BARONE L, et al. A Faster algorithm for calculating hypervolume[J]. IEEE Transactions on Evolutionary Computation, 2006, 10(1): 29-38.
- [4] BEUME N, RUDLOPH G. Faster Smetric Calculation by Considering Dominated Hyper volume as Klee's Measure problem [C]// Proc. of the 2nd IASTED Conference on Computational Intelligence. Anaheim, USA: ACTA Press, 2006.
- [5] ZHANG Y, ZHANG M, LIANG Y C. Application of an Improved Dijkstra Algorithm in Multicast Routing Problem[J]. Computer Science, 2009, 36(8): 205-207. (in Chinese)
- [6] DONG W, ZHANG D, WEICHENG Z, et al. The Simulation Optimization Algorithm Based on the Ito Process[C]// Advanced Intelligent Computing Theories and Applications with

- Aspects of Contemporary Intelligent Computing Techniques. 2007;115-124.
- [7] DONG W, LEI M, YU R. A New Evolutionary Algorithms for Global Numerical Optimization Based on Ito Process[M]//Computational Intelligence and Intelligent Systems; 5th International Symposium (ISICA 2010). Wuhan, China, Springer, Berlin, Heidelberg, 2010.
- [8] DONG W, YU R, LEI M. Merging the Ranking and Selection into ITO Algorithm for Simulation Optimization[C]//5th International Symposium Computational Intelligence and Intelligent Systems (ISICA 2010). Wuhan, China, Springer, 2010.
- [9] DONG W Y, ZHANG W S, YU R G. Convergence and Runtime Analysis of ITO Algorithm for One Class of Combinatorial Optimization[J]. Chinese Journal of Computers, 2011, 34(4): 636-646. (in Chinese)
董文永, 张文生, 于瑞国. 求解组合优化问题伊藤算法的收敛性和期望收敛速度分析[J]. 计算机学报, 2011, 34(4): 636-646.
- [10] YI Y F, CAI Y L, DONG W Y, et al. Improved ITO Algorithm for Multiobjective Real-time Vehicle Routing Problem with Customers' Satisfaction[J]. Acta Electronica Sinica, 2015, 43(10): 2053-2061. (in Chinese)
易云飞, 蔡永乐, 董文永, 等. 求解带用户满意度的多目标实时车辆路径问题的改进伊藤算法[J]. 电子学报, 2015, 43(10): 2053-2061.
- [11] YI Y F, DONG W Y, LIN X D, et al. The Improved ITO Algorithm to Solve the Vehicle Routing Problem with Soft Time Windows and Its Convergence Analysis[J]. Acta Electronica Sinica, 2015, 43(4): 658-664. (in Chinese)
易云飞, 董文永, 林晓东, 等. 求解带软时间窗车辆路径问题的改进伊藤算法及其收敛性分析[J]. 电子学报, 2015, 43(4): 658-664.
- [12] WANG H G, YU S M. Improved ITO Algorithm for Solving VRP[J]. Computer Science, 2015, 42(9): 253-256. (in Chinese)
王浩光, 余世明. 求解车辆路径问题的改进伊藤算法[J]. 计算机科学, 2015, 42(9): 253-256
- [13] ZHAO Z Y, LI Y X, YU F. Artificial Bee Colony Algorithm Based on ITO Algorithm[J]. Computer Science, 2014, 41(6A): 29-32. (in Chinese)
赵志勇, 李元香, 喻飞. 基于伊藤算法的改进人工蜂群算法[J]. 计算机科学, 2014, 41(6A): 29-32.
- [14] YI Y F, CAI Y L, DONG W Y, et al. Improved ITO Algorithm for Solving the CVRP[J]. Computer Science, 2013, 40(5): 213-216. (in Chinese)
易云飞, 蔡永乐, 董文永, 等. 求解带容量约束的车辆路径问题的改进伊藤算法[J]. 计算机科学, 2013, 40(5): 213-216.
- [15] MUNEMOTO M, TAKAI Y, SATO Y. A Migration Scheme for the Genetic Adaptive Routing Algorithm[C]//IEEE International Conference on Systems, Man, and Cybernetics. 1998: 2774-2779.
- [16] INAGAKI J, HASEYAMA M, KITAJIMA H. A Genetic Algorithm for Determining Multiple Routes and Its Applications[C]//Proceedings of IEEE International Symposium on Circuits and Systems. 1999: 137-140.
- [17] AHN C W, RAMAKRISHNA R S. A Genetic Algorithm for Shortest Path Routing Problem and the Sizing of Populations[J]. IEEE Transactions on Evolutionary Computation, 2002, 6(6): 566-579.
- [18] SUN B L, LI L Y, CHEN H. Shortest Path Routing Optimization Algorithms Based on Genetic Algorithms[J]. Computer Engineering, 2005, 31(6): 142-144. (in Chinese)
孙宝林, 李腊元, 陈华. 基于遗传算法的最短路径路由优化算法[J]. 计算机工程, 2005, 31(6): 142-144.

(上接第 196 页)

- [7] ZHANG Y, LU R, YANG Q. Predicting retweeting in microblogs[J]. Journal of Chinese Information Processing, 2012, 26(4): 109-114. (in Chinese)
张畅, 路荣, 杨青. 微博客中转发行为的预测研究[J]. 中文信息学报, 2012, 26(4): 109-114.
- [8] LI Y L, YU H T, LIU L X. Predict algorithm of micro-blog retweet scale based on SVM[J]. Application Research of Computers, 2013, 30(9): 2594-2597. (in Chinese)
李英乐, 于洪涛, 刘力雄. 基于 SVM 的微博转发规模预测方法[J]. 计算机应用研究, 2013, 30(9): 2594-2597.
- [9] ZHAO Y, SHAO B L, BIAN G Q, et al. Prediction of retweeting behavior for imbalanced dataset in microblogs[J]. Journal of Computer Applications, 2015, 35(7): 1959-1964. (in Chinese)
赵煜, 邵必林, 边根庆, 等. 面向不平衡微博数据集的转发行为预测方法[J]. 计算机应用, 2015, 35(7): 1959-1964.
- [10] SUH B, HONG L, PIROLI P, et al. Want to be Retweeted? Large scale analytics on factors impacting retweet in Twitter network[C]//2010 IEEE International Conference on Social Computing / IEEE International Conference on Privacy, Security, Risk and Trust. IEEE, 2010: 177-184.
- [11] HU W. Real-time Twitter sentiment toward midterm exams[J]. Sociology Mind, 2012, 2(2): 177-184.
- [12] WU J H, ZUO K Z, JIE B, et al. New discriminative feature selection method[J]. Journal of Computer Applications, 2015, 35(10): 2752-2756. (in Chinese)
吴锦华, 左开中, 接标, 等. 新颖的判别性特征选择方法[J]. 计算机应用, 2015, 35(10): 2752-2756.
- [13] KAR M, NUNES S, RIBEIRO C. Summarization of changes in dynamic text collections using Latent Dirichlet Allocation model[J]. Information Processing & Management, 2015, 51(6): 809-833.
- [14] LIU S P, YIN J, OUYANG J, et al. Topic mining from microblogs based on MB-HDP model[J]. Chinese Journal of Computers, 2015(7): 1408-1419. (in Chinese)
刘少鹏, 印鉴, 欧阳佳, 等. 基于 MB-HDP 模型的微博主题挖掘[J]. 计算机学报, 2015(7): 1408-1419.