

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

易云飞^{1,2,3},董文永¹,林晓东²,蔡永乐¹

(1.武汉大学计算机学院,湖北武汉 430079;2.河池学院计算机与信息工程学院,广西宜州 546300;
3.广西混杂计算与集成电路设计分析重点实验室,广西南宁 530006)

摘要: 针对伊藤算法在求解离散组合优化问题时效率较低、收敛性较差等缺陷,本文提出的改进伊藤算法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确定波动系数,将漂移和波动同步进行,当找到可行解之后再进行一次程度的波动.为了验证算法的有效性,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.仿真结果表明,改进后的算法效率更高,收敛速度更快,算法稳定性和健壮性也更好.此外,本文还根据马尔科夫链移向吸引元的性质及其各状态之间的转换关系,探讨了构造伊藤随机微分方程的马尔科夫链近似模拟算法及其收敛性证明.

关键词: 伊藤算法; 漂移算子; 波动算子; 收敛性分析; 带软时间窗车辆路径问题

中图分类号: TP301 **文献标识码:** A **文章编号:** 0372-2112 (2015)04-0658-07

电子学报 URL: <http://www.ejournal.org.cn> **DOI:** 10.3969/j.issn.0372-2112.2015.04.006

The Improved ITO Algorithm to Solve the Vehicle Routing Problem with Soft Time Windows and Its Convergence Analysis

YI Yun-fei^{1,2,3}, DONG Wen-yong¹, LIN Xiao-dong², CAI Yong-le¹

(1. Computer School, Wuhan University, Wuhan, Hubei 430079, China;

2. College of Computer and Information Engineering, Hechi University, Yizhou, Guangxi 546300, China;

3. Guangxi Key Laboratory of Hybrid Computation and IC Design Analysis, Nanning, Guangxi 530006, China)

Abstract: In order to overcome the shortcoming of the efficiency of the Ito algorithm for the discrete combinatorial optimization problems, an improved Ito algorithm based on collaborative diffusion coefficient and Hill climbing was proposed in this paper. To be consistent with the principle of Brownian motion, the drift and wave should be moved simultaneously, when it found a feasible solution, which continued to be a degree of renewed volatility. Experimental results show that the improved Ito algorithm for solving vehicle routing problem with a soft time window is valid, with the convergence speed, robustness and stability, especially Ito algorithm combines the ability of local search algorithms after climbing method performance has been greatly improved. Finally, according the nature of Markov chain to move to its attractive element and the transformation of the relationship between each state, the Markov chain approximation simulation algorithm which structures the Ito stochastic differential equation and its convergence was proved in this paper.

Key words: Ito algorithms; drift operator; wave operator; convergence analysis; vehicle routing problem with soft time windows

1 引言

由 G Dantzig 和 J Ramser 于 1959 年首次提出了车辆路径问题^[1](Vehicle Routing Problem, VRP).之后,研究者对问题进行扩展后提出了诸如带容量约束的^[2]、带时间窗的^[3]、循环取货的^[4]、集货和配送可拆分的^[5]车辆路径问题.求解这些问题的方法也有很多种,而最常用的

是元启发式算法,主要包括:模拟退火算法^[6]、禁忌搜索算法^[7]、遗传算法^[8]、蚁群算法^[9]、粒子群算法^[10,11]等.然而,元启发式算法在实践的过程中是一种通用的优化器,普遍存在通用与效率、探索与开发、效率与精度等方面的矛盾,急需一种新型的计算模型和理论作为支撑.基于此,武汉大学董文永教授和李元香教授首次提出了伊藤算法(Ito Algorithm, ITO).

伊藤算法^[12-16]一方面体现了仿生演化算法群体搜索的特征,解是用粒子表示的,大量的粒子组成粒子系统;另一方面运用了伊藤随机过程的理论来分析算法,利用伊藤随机积分的方法建立算法的动力学方程.算法中的漂移和波动算子是全局寻优性能的关键点.

鉴于经典 VRP 问题本身的扩展,本文用于实验验证的是参考文献[3]的带软时间窗的车辆路径问题(Vehicle Routing Problem With Soft Time Windows, VRPSTW).为了更有效地求解该问题,本文对伊藤算法进行了改进,实验结果表明改进后的算法效果更好.此外,本文还进行了收敛性分析和证明.

2 带软时间窗车辆路径问题模型

VRP^[1]的描述为:在客户网络里,已知仓库和待服务的客户的位置、各客户的需求及车辆的最大负荷,要求以仓库为起点和终点设计出若干条合适的车辆路线从而保证各客户只被其中的一辆车访问过一次,并使总的运输成本最小.VRPSTW 增加了时间窗约束(各客户给出自己接受服务的时间段),目标函数是变为行程总费用(车辆启用固定费用+路径费用+惩罚费用+服务费用)最少.带软时间窗的车辆路径问题可以转化成图模型,其中,最基本的要素是这个图 $G(V, E)$ 的边集合.结合图论,VRPSTW 的基本数学模型描述如下:

$$f_{\min} = NV * VecCost + PC * \sum_{i=0}^N \sum_{j=0}^N \sum_{k=1}^{NV} c_{ij} x_{ijk} + Cw * \sum_{i=1}^N wt_i + Cs * \sum_{i=1}^N st_i \quad (1)$$

$$x_{ijk} = \begin{cases} 1, & \text{车辆 } k \text{ 从点 } i \text{ 行驶到点 } j \\ 0, & \text{否则} \end{cases} \quad (i, j = 0, 1, \dots, N; k = 1, 2, \dots, NV) \quad (2)$$

$$y_{ki} = \begin{cases} 1, & \text{点 } i \text{ 的任务由车辆 } k \text{ 完成} \\ 0, & \text{否则} \end{cases} \quad (i = 0, 1, \dots, N; k = 1, 2, \dots, K) \quad (3)$$

$$\sum_i q_i y_{ki} \leq Q \quad (i = 0, 1, \dots, N; k = 1, 2, \dots, K) \quad (4)$$

$$\sum_{k=0}^{NV} y_{ki} = 1 \quad (i = 0, 1, \dots, N; k = 1, 2, \dots, K) \quad (5)$$

$$\sum_{j=0}^N x_{ijk} = y_{ki} \quad (i = 1, 2, \dots, N; k = 1, 2, \dots, NV) \quad (6)$$

$$\sum_{i=0}^N x_{ijk} = y_{kj} \quad (j = 0, 1, 2, \dots, N; k = 1, 2, \dots, NV) \quad (7)$$

$$CurTime \in [e_i, l_i] \quad (8)$$

在该模型中,仓库编号为 0,客户需求点编号为 1, 2, 3, ..., N, 车速 Speed, 车辆启用固定费用 VecCost, 车辆数 NV, 单位路径成本 PC, 客户点的时间窗 $[e_i, l_i]$ (其中 e_i 为最早开始服务时间, l_i 为最迟开始服务时间),

服务客户点 i 的等待时间 wt_i , 等待时间单位成本 Cw , 服务客户点 i 的服务时间 st_i , 服务时间单位成本 Cs , 到达下一个客户点的时间为 CurTime, 任意两点之间的距离为 c_{ij} .

其中,式(1)为目标函数,表示完成所有客户点的配送任务花费的总费用;式(2)定义 x_{ijk} , 表示车辆 k 是否经过客户点 i 行驶到客户点 j ;式(3)定义 y_{ki} , 表示客户点 i 是否由车辆 k 提供服务;式(4)~式(8)为约束条件:式(4)表示车辆的容量约束,每条行车路线上的客户需求量总和不能超过车辆的最大载货量;式(5)表示每个客户需求点都要被车辆访问过,而且只能被一个车辆访问过;式(6)和(7)表示在车辆 k 行驶路线上的点 i, j 都是要被车辆 k 访问的点;式(8)表示时间约束,不满足就在目标函数中额外增加等待费用以示惩罚.

3 改进伊藤算法求解带软时间窗车辆路径问题

本文首先要解决的问题是将伊藤随机过程映射为优化算法:一个是如何设计伊藤过程的宏观趋势项,即漂移过程,使得算法在宏观上朝着最优解的方向前进;另一个是如何设计伊藤过程的随机波动项,即波动过程,使得粒子在自身局部空间内进行搜索.这两个算子的设计与粒子的半径和温度密切相关.有效的设计漂移和波动算子对算法的性能来说是非常重要的.

3.1 冷却进度表的设计

按照爱因斯坦和 P 朗之万的理想气体巨分子系统的位移方程,粒子的漂移和波动随温度的变化而变化,即:在高温条件下,粒子的能量较高,剧烈运动;在低温条件下,粒子的能量较低,运动能力受到束缚.可见,环境温度的变化可以影响到群体中粒子的漂移和波动,有效的设计环境温度能促进粒子搜索到全局最优解.可行的方案之一是采用模拟退火算法中的设计方法,用冷却进度表作为控制温度变化的参数,用其逼近算法的渐进收敛形态,使算法在有限时间的执行后返回一个近似最优解.退火函数设置如下:

$$T_t = \rho * T_{t-1} \quad (9)$$

其中, $0 \leq \rho \leq 1$, 一般情况下取 $\rho = 0.99$.

3.2 半径的设计

搜索解空间中的每一个粒子都有一个半径属性,伊藤算法中的粒子半径会根据粒子的当前状态动态地变化,从而影响了粒子漂移和波动的剧烈程度.半径越大的粒子运动能力越小,半径越小的粒子运动能力越剧烈.于是,针对极小化问题,目标函数值较大的粒子半径较小,从而提高运动能力,在大范围内运动达到搜索效果;对于极大化问题,目标函数值较大的粒子半径较大,能使粒子稳定在周围的邻域中搜索,提高算法的

局部搜索能力,防止算法陷入局部最优解.有很多函数都可以用来设计粒子的半径,例如,线性变换函数,指数函数等.本文采用基于排序的粒子半径计算方法,计算步骤如下:

首先,对种群中的所有粒子按照目标函数值进行降序排序 $\{n_1, n_2, n_3, \dots, n_M\}$, 其中 $n_1, n_2, n_3, \dots, n_M$ 代表 M 个粒子.

然后,按式(10)计算每个粒子的半径.

$$r_{n_i} = r_{\min} + \frac{(r_{\max} - r_{\min})}{M} * i (i \in [0, M]) \quad (10)$$

其中 r_{\max}, r_{\min} 分别为半径的上下界, M 为粒子总数. 本文假设 $r_{\min} = 0, r_{\max} = 1$. 于是,式(10)化为:

$$r_{n_i} = \frac{1.0}{M} * i (i \in [0, M]) \quad (11)$$

3.3 漂移算子的设计

漂移算子^[14]主要是用来控制粒子在宏观上的运动.在设计漂移算子时需要考虑到漂移的上界和下界,避免出现无效解.一般漂移算子的设计需要考虑两个方面:一个是漂移强度,即漂移的系数 μ ;另一个是漂移的过程,即粒子如何按照漂移强度来完成在邻域内的漂移.漂移系数 μ 与粒子自身的半径 r ,当前的环境温度 T 等因素有关.因此,可以设计一个可分离变量的函数如下:

$$f(r, T) = \mu_{\min} + f_1(r) * f_2(T) * (\mu_{\max} - \mu_{\min}) \quad (12)$$

其中, μ_{\max}, μ_{\min} 分别为粒子的最大漂移强度和最小漂移强度.为了使算法设计方便,本文中设置 $\mu_{\max} = 1, \mu_{\min} = 0$. 函数 $f_1(r)$ 为粒子半径决定的漂移强度,一种可行的方案为:

$$f_1(r) = \frac{(e^{-\lambda r} - e^{-\lambda r_{\max}})}{(e^{-\lambda r_{\min}} - e^{-\lambda r_{\max}})} \quad (13)$$

$$\tau(i, j) + = \begin{cases} 0.85 + \mu_g, & \text{其他(向目前全局最优解的方向漂移)} \\ 0.9 + \mu, & \text{if } e(i, j) \in \sigma \text{(保留当前粒子的运动趋势)} \\ 0.95 + \mu_p, & \text{if } e(i, j) \in \sigma_p \text{(向当前代的最优解方向漂移)} \\ 0.6 + \mu_p + \gamma, & \text{if } e(i, j) \in \sigma_p \cap e(i, j) \in \sigma \text{(保留原来的运动趋势和向当前代的最优解漂移)} \end{cases} \quad (17)$$

其次,当粒子以式(17)更新完各边权重之后,再以当代和目前最优解来更新权重信息以提高两个最优解对群体的影响力,使两个极值带动群体向更好的方向搜索,从而达到搜索全局最优解的效果.更新方式为:

$$\tau(i, j) + = \begin{cases} 1.2 + \gamma_g, & \text{if } e(i, j) \in \sigma_g \\ 0.9 + \gamma_p + \mu_g, & \text{if } e(i, j) \in \sigma_p \cap e(i, j) \in \sigma_g \end{cases} \quad (18)$$

其中, $\tau(i, j)$ 表示在边 (i, j) 上的权重,初值为一个常数 C ; $\sigma, \sigma_p, \sigma_g$ 分别表示当前粒子,当代最优粒子,目前最优粒子的路径; μ, μ_p, μ_g 分别表示当前粒子,当代最优粒子,目前最优粒子的漂移强度; $\gamma, \gamma_p, \gamma_g$ 分别表

函数 $f_2(T)$ 表示粒子受环境温度的影响,一种可行的设计方案是采用模拟退火算法中的 Metropolis 准则,其结构如下:

$$f_2(T) = \exp\left(-\frac{1}{T}\right) \quad (14)$$

其中, T 是当前的环境温度. 于是:

$$f(r, T) = \frac{(e^{-\lambda r} - e^{-\lambda})}{(1 - e^{-\lambda})} * \exp^{-1/T} \quad (15)$$

3.4 波动算子的设计

波动算子^[16]是用来控制粒子在微观上的运动,提高粒子的探索特性,使粒子在邻域内局部扰动.其中波动算子也与漂移算子一样有两方面的问题需要考虑:一是波动强度,也即波动的系数 γ ;另一个是波动的过程,即粒子如何按照波动强度完成在邻域内的有效波动.针对漂移算子在宏观上朝着吸引元的方向快速漂移,对局部的搜索能力较差.所以,本文首先将波动算子设计为:

$$f_\gamma = \mu * f_2(T) \quad (16)$$

有了波动强度,粒子将按照该波动强度与漂移强度同步进行和各个粒子之间协同扩散搜索空间中的全局最优解,避免在漂移过程中缺乏波动性而导致算法陷入局部最优解,搜索不到全局最优解.

3.5 漂移、波动算子的改进设计

伊藤算法中,漂移过程和波动过程往往是单独进行的.而在布朗运动中,粒子在向吸引元方向移动的过程中,漂移和波动往往是同步进行的,当达到吸引元的周围时,粒子不是静止不动而是处于不断的波动中,这更加符合布朗运动规律.基于此,本文提出将漂移和波动同步进行再波动的设计方法.首先对粒子漂移和波动同时进行操作,权重更新方式为:

示当前粒子,当代最优粒子,目前最优粒子的波动强度.

更新完各边的权重之后,根据马尔科夫链各状态之间的转换关系依据各边权重的相关信息采用轮盘赌的方案来更新群体,使新一代向全局最优解的方向移动,防止出现退化现象.概率选取公式设计为:

$$\rho(i, j) = \begin{cases} \frac{[\tau(i, j)]^\alpha [\eta(i, j)]^\beta}{\sum_{t \in \text{tabu}_k} [\tau(i, t)]^\alpha [\eta(i, t)]^\beta}, & i \in \text{tabu}_k \cap j \notin \text{tabu}_k \\ 0, & \text{其他} \end{cases} \quad (19)$$

其中, $\rho(i, j)$ 表示路径 (i, j) 被选中的概率, $\rho(i, j)$

越大,路径 (i, j) 被选中的概率越大; α 表示控制边权重在概率选择中重要程度的因子; β 为控制能见度(边长因素)在概率选择中重要程度的因子; $\eta(i, j)$ 为距离的倒数,即 $1/c_{ij}$,称为经验知识; $tabu_k$ 记录粒子 k 当前所走过的路径,称为禁忌表。

新一代群体产生后,基于粒子在运动过程中总是处于运动状态的思想,本文采用局部搜索能力很强的爬山算法来对算法进行再次波动,增加算法的局部搜索能力,使粒子尽可能保持多样性,避免陷入局部最优。

3.6 用改进伊藤算法求解带软时间窗车辆路径问题的算法设计

算法流程如下:

Step1 根据 VRPSTW 来初始化算法中的各项参数,并使 $\tau(i, j)$ 的初值为 1。

Step2 采用式(19)来更新所有粒子的路径,但在更新路径时是一边构造路径一边判断是否满足约束条件,若满足,则将此客户点加到路径中去再计算当前的适应度值,当构造完所有粒子的路径时更新当代最优解和全局最优解。

Step3 根据式(9)和(11)更新粒子群当前的环境

温度和所有粒子的半径。

Step4 根据式(15)和(16)来设计伊藤算法中的漂移算子和波动算子,从而依据式(17)和(18)来更新各条边的权重信息。

Step5 更新各条边的权重信息完之后,由于粒子时时刻刻处于运动中,所以对其进行再次的波动并更新每个粒子的最优解。

Step6 对新一代群体按照适应度值进行从大到小排序并更新全局最优解。

Step7 判断是否满足停机条件,若满足,则算法结束,此时记录的全局最优解就是算法找到的最优解;否则返回到 Step3 继续搜索新一代的种群。

4 实验及其结果分析

为了验证改进算法的有效性,本文采用 Solomon Benchmark 标准测试集 r101 ~ r112(50)对文献[12]中的伊藤算法和本文提出的改进伊藤算法分别进行数据测试,相关参数设置见表 1,实验结果见表 2.实验仿真环境: Intel(R) Core(TM) i3 M 370 @2.40GHz, 2GB 内存, Win7 系统.实验仿真软件: Eclipse Helios Service Release 2.

表 1 实验中各项参数的设置

参数	值	参数	值	参数	值	参数	值
种群大小 M	50	退火表初始温度 T	1000	初始权重	1	退火表长度 $TLength$	5
车速 $Speed$	1	漂移用到的参数 λ	1.4	最大进化代数 GEN	150	客户点之间距离重要性因子 β	2
客户数 N	50	边权重重要因子 α	5	车费用 $VecCost$	500	单位等待时间惩罚费用 Cw	10
停机条件	> GEN	退火速率 ρ	0.99	单位路径成本 PC	40	单位时间服务费用 Cs	5

表 2 R101 ~ R112 数据对比

Pro.	Distance			NV			Deviation	
	ITO	改进 ITO	Best Known	ITO	改进 ITO	Best Known	ITO	改进 ITO
r101	1085.2	1061.1	1044.0	12	12	12	3.9%	1.6%
r102	1032.9	971.1	909	10	10	11	13.6%	6.8%
r103	851.7	819.6	772.9	9	8	9	10.2%	6.0%
r104	727.7	670.4	625.4	6	6	6	16.4%	7.2%
r105	940.3	939.4	899.3	10	9	9	4.6%	4.5%
r106	897.7	893.8	793	9	8	5	13.2%	12.7%
r107	807.0	772.2	711.1	7	7	7	13.5%	8.6%
r108	697.7	653.3	617.7	6	6	6	13.0%	5.8%
r109	867.3	832.8	786.8	8	8	8	10.2%	5.8%
r110	762.7	752.1	697.0	8	7	7	9.4%	7.9%
r111	819.1	767.7	707.2	7	7	7	15.8%	8.6%
r112	716.2	678.9	630.2	6	6	6	13.6%	7.7%

通过对本文改进的伊藤算法进行多轮数据测试,以下对 r103 进行分析:得出最优总费用 42377.346,对应的路径长度 819.564;最差总费用 43272.264,对应的

路径长度 848.788;平均总费用 43061.577,平均路径长度 840.526.虽然,本文所得出的结果没有优于目前所知道的最优解(Solomon Benchmark 测试集 R103.50,车辆数

$NV = 9$, 路径长度 $Distance = 772.9$), 但却节约一辆车的费用, 这大大节约了人力和物力在各方面的费用. 而标准伊藤算法求解 VRPSTW 得出的结果为: 最优总费用 45655.271, 对应的路径长度 851.708; 最差总费用 46847.335, 对应的路径长度 894.014; 平均总费用 46501.657, 平均路径长度 873.470. 显然, 改进后的算法在求解 VRPSTW 中更有效.

伊藤算法改进前后求解 VRPSTW 的仿真结果及性能比较如图 1 - 图 8 所示. 其中, 图 1 和图 4 分别为改进前和改进后的实验结果最优总费用对应的路径的仿真; 图 2 和图 5 分别为该算法在改进前和改进后总费用的全局收敛速度; 图 3 和图 6 分别为该算法在改进前和改进后运行 20 次总费用最优对应的路径的结果图. 从图 8 显示的总费用全局收敛速度对比可知, 本文设计的伊藤算法在全局收敛上速度快, 在迭代到 70 代左右时已经收敛到全局最优解. 从而说明, 改进后的 ITO 能够有效地求解 VRPSTW 问题. 从图 7 显示的运行 20 次的实验结果图对比可以看出, 本文算法稳定性高, 大部分结果处于 840 上下范围之内, 而 ITO 改进前实验结果处于 850 - 900 之间且稳定性较差. 显然, 改进后的算法是可行的, 并且具有较强稳定性, 健壮性和快速收敛等特点.

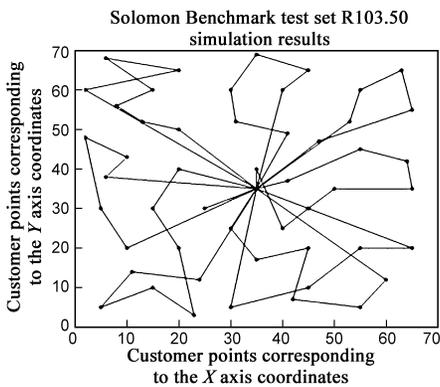


图1 用伊藤算法求解的仿真结果

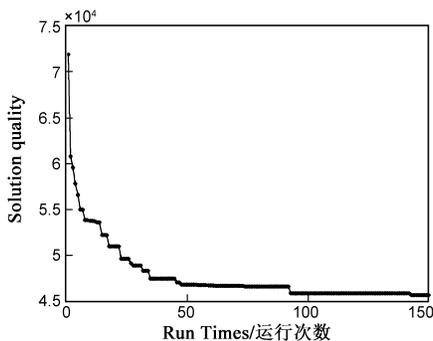


图2 伊藤算法的全局收敛速度

仍存在不足之处, 如 R106.50 类与最优解偏差较大. 除了实例 R102 和 R103 比最优解少了一辆车, R106 比最优解多 3 辆车外, 其他实例的车辆数均与最优解相同. 与 ITO 算法相比, 其所得结果均优, 而且在 R103、R105、R106、R110 实例中减少一辆车费用. 与最优解相比 (除 R106), 偏差都控制在 8.6% 之内. 总体来说, 本文算法在求解 VRPSTW 是可行的, 今后将对这些实例求解存在的问题做进一步的研究.

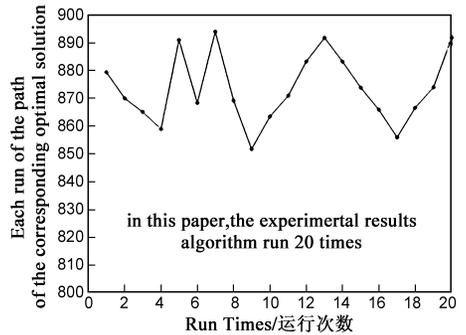


图3 伊藤算法运行20次的实验结果

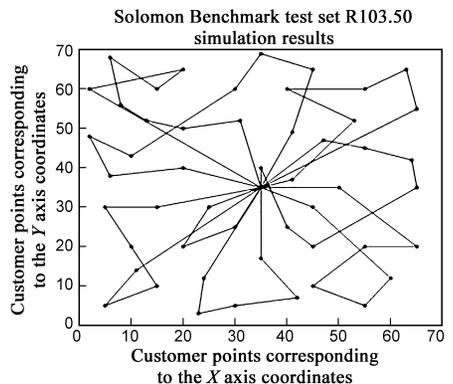


图4 用本文伊藤算法求解的仿真结果

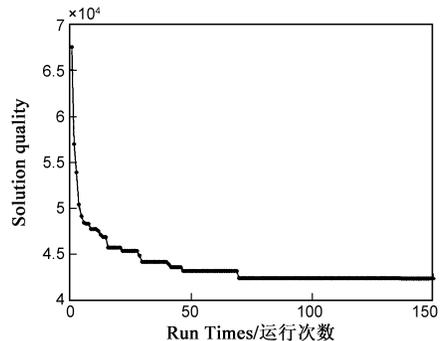


图5 本文伊藤算法的全局收敛速度

5 算法的收敛性分析

通过以上的详细分析与设计, 可以看出改进的伊藤算法的有效性. 为了证明改进后伊藤算法的全局收敛性, 本文将伊藤算法的迭代过程映射为 Markov 链, 然

然而, 从表 2 中可以看出, 本文算法在求某些问题

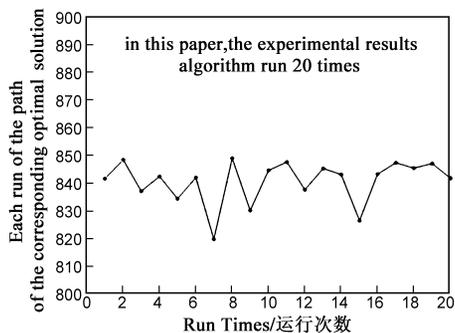


图6 本文伊藤算法运行20次的结果

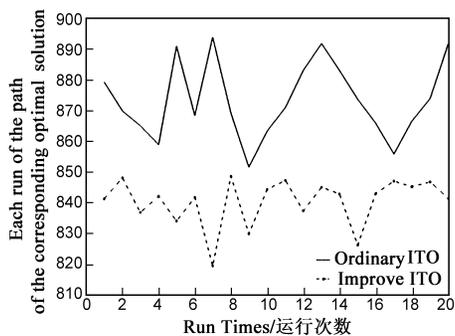


图7 改进前后伊藤算法求得结果比较

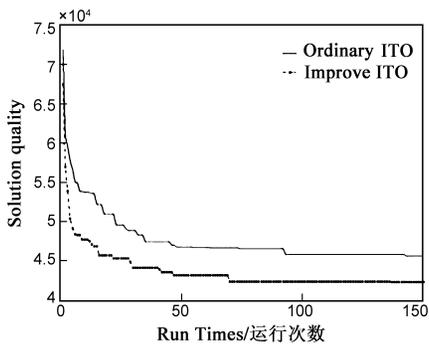


图8 改进前后伊藤算法收敛性比较

后采用 Markov 链来分析算法的全局收敛性。根据马尔科夫链移向吸引元的性质及其各状态之间的转换关系,探讨了构造伊藤随机微分方程的马尔科夫链近似模拟算法及其收敛性证明。根据伊藤积分公式,可以将漂移过程理解为公式第一项,即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,伊藤方程中的漂移就可以映射到粒子的向优性上,再对其进行波动。具体证明过程如下:

定义 1 设有随机过程 $X_T = \{X_t, t \in T = (0, 1, 2, \dots)\}$ 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:

$$P(X_{t_0} = i_0, X_{t_1} = i_1, \dots, X_{t_k} = i_k) > 0 \quad (20)$$

总有,

$$P(X_{t_{k+1}} = i_{k+1} | X_{t_0} = i_0, X_{t_1} = i_1, \dots, X_{t_k} = i_k) = \quad (21)$$

$$P(X_{t_{k+1}} = i_{k+1} | X_{t_k} = i_k)$$

成立,则称 X_T 为离散时间的 Markov 链。

定理 1 伊藤算法的运行过程是一时间齐次的有限马尔科夫链。

证明 从算法的描述上来看,伊藤算法中的每一个粒子都是按照自身的运动规律进行运动并根据当前的两个极值来引导自身运动,但为了分析算法整体的运行过程,先要从分析一个粒子入手。如果把时刻 t_k 看做现在,那么 t_{k+1} 是将来的时刻,而 t_0, t_1, \dots, t_{k-1} 则是过去的时刻。假设粒子在时刻 t_k 处于状态 i_k 的条件下,那么在时刻 t_{k+1} 处于的状态是由状态 i_k , 漂移算子,波动算子,温度,半径,吸引元等因素决定的,而与过去的时刻是无关的。所以从伊藤算法的整体上来看,伊藤算法的第 $X_{t_{k+1}}$ 代种群只取决于第 X_{t_k} 代种群,而与过去的其他时刻所处的状态 $X_{t_0}, X_{t_1}, \dots, X_{t_{k-1}}$ 的种群是没有关系,并且每一代种群的转移概率互不相同。因此,伊藤算法的搜索过程是一时间齐次的有限马尔科夫链。

证毕。

引理 1 有限齐次马尔科夫链从任意非常返状态出发以概率 1 必定要到达常返状态。

引理 2 状态空间 Ω 必可分解为 $\Omega = N + C_1 + C_2 + \dots + C_n + \dots$ 。其中, N 是全部非常返态组成的集合, $C_1, C_2, \dots, C_n, \dots$ 是互不相交的由常返态组成的闭集,同时还满足:

(i) 对于每一确定的 n, C_n 内任意两状态是相通的。

(ii) C_n 与 $C_g (n \neq g)$ 中的状态之间不相通。

定理 2 求解带软时间窗车辆路径问题的改进伊藤算法最终以概率 1 收敛到全局最优解。

证明 设 Ω^* 为所有包含全局最优解的子群体的集合。由马尔科夫链理论可以得知, $(\Omega - \Omega^*)$ 中的状态是一个非常返状态,而 Ω^* 中的状态是一个常返状态,则由引理 1 和引理 2 知,若从某个非常返状态出发,则粒子可能在非常返集合 N 中运动,也可能在某个时刻进入到某个闭集中运动,从此就一直在该闭集中运动,若从某一常返状态出发,则粒子就永远在该常返状态所在的闭集中运动。因此,无论伊藤算法中的粒子是从非常返集合中出发还是从常返集合中出发,最终都能以概率 1 收敛到全局最优解。

证毕。

6 结论

为了使得伊藤算法更加符合布朗运动规律,本文

在传统的伊藤算法的基础上做出了改进,将漂移和波动同步进行,当找到一个可行解的时候仍然进行不同程度的再次波动.此外,文中还引入了爬山法对波动算子设计进行改进,引入轮盘赌来改进路径的选择.之后,用改进后的伊藤算法求解带软时间窗车辆路径问题进行试验验证,结果显示了算法的良好性能.最后,本文还证明了改进后的伊藤算法以概率 1 收敛到全局最优解,并对求解 VRPSTW 的改进伊藤算法进行了收敛性分析.很显然,待求解问题的规模和复杂度都可以加大,文中的参数还可以进一步调整,进而提升算法的效率和精度.此外,在设定算法的参数时,如何做到更加科学、合理,而不是凭经验或者多次尝试,这些将是进一步研究的方向.

参考文献

- [1] Dantzig G B, Ramser J H. The truck dispatching problem[J]. *Management Science*, 1959, 6(1): 80 – 91.
- [2] Szeto W Y, Wu Y, et al. An artificial bee colony algorithm for the capacitated vehicle routing problem[J]. *European Journal of Operational Research*, 2011, 215(1): 126 – 135.
- [3] Gong Y J, Zhang J, et al. Optimizing the vehicle routing problem with time windows: a discrete particle swarm optimization approach[J]. *IEEE Trans Syst*, 2012, 42(2): 254 – 267.
- [4] KONG Ji-li, JIA Guo-zhu, et al. A new mathematical model of vehicle routing problem based on milk-run [A]. *International Conference on Management Science & Engineering [C]*. Harbin: IEEE press, 2013. 385 – 392.
- [5] Wang Y, Ma X L, et al. A two-stage heuristic method for vehicle routing problem with split deliveries and pickups[J]. *Journal of Zhejiang University-Science C*, 2014, 15(3): 200 – 210.
- [6] Chiang W C, Russell R A. Simulated annealing metaheuristics for the vehicle routing problem with time windows[J]. *Ann Oper Res*, 1996, 63(1): 3 – 27.
- [7] Taillard E, Badeau P, et al. A tabu search heuristic for the vehicle routing problem with soft time windows[J]. *Transp Sci*, 1997, 31(2): 170 – 186.
- [8] Ghoseiri K, Ghannadpour S F. Hybrid genetic algorithm for vehicle routing and scheduling problem[J]. *Appl Sci*, 2009, 9(1): 79 – 87.
- [9] 吴兆福, 董文永. 求解动态车辆路径问题的演化蚁群算法[J]. *武汉大学学报(理学版)*, 2007, 53(5): 571 – 575.
Wu Zhaofu, Dong Wenyong. A mixed evolutionary ant algorithm for the dynamic vehicle routing problem[J]. *Wuhan Univ*, 2007, 53(5): 571 – 575. (in Chinese)
- [10] Lin C J, Chen C H, et al. A hybrid of cooperative particle swarm optimization and cultural algorithm for neural fuzzy networks and its prediction applications[J]. *IEEE Trans Syst Man, Cybern C, Appl Rev*, 2009, 39(1): 55 – 68.
- [11] 喻飞, 李元香, 等. 透镜成像反学习策略在粒子群算法中的应用[J]. *电子学报*, 2014, 42(2): 230 – 235.
YU Fei, LI Yuan-xiang, et al. The application of a novel OBL based on lens imaging principle in PSO[J]. *Acta Electronica Sinica*, 2014, 42(2): 230 – 235. (in Chinese)
- [12] Dong Wenyong, Zhang Dengyi. Simulation optimization based on the hypothesis testing and ITO process[A]. *Third International Conference on Natural Computation [C]*. Haikou: IEEE Press, 2007. 660 – 665.
- [13] Dong Wenyong, Yu Ruiguo, et al. Merging the ranking and selection into ITO algorithm for simulation optimization[A]. *International Conference on Intelligence Computation and Applications [C]*. Wuhan: Springer-Verlag Berlin Heidelberg, 2010. 87 – 96.
- [14] Dong Wenyong, Lei Ming, et al. A new evolutionary algorithms for global numerical optimization based on ITO process [A]. *International Conference on Intelligence Computation and Applications [C]*. Wuhan: Springer-Verlag Berlin Heidelberg, 2010. 57 – 67.
- [15] 董文永, 张文生, 等. 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.
DONG Wen-yong, ZHANG Wen-Sheng, et al. 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)
- [16] Wenyong DONG, Ming LEI, et al. BBOB-benchmarking: a new evolutionary algorithms inspired by ITO process for noiseless function testbed[J]. *Journal of Computational Information Systems*, 2011, 7(6): 2195 – 2203.

作者简介



易云飞 男, 1981 年出生于广西资源, 武汉大学计算机学院博士生, 副教授, 研究方向为智能计算、机器学习等。
E-mail: gxyiyf@163.com



董文永 男, 1973 年出生于河南南阳, 武汉大学计算机学院教授, 博士生导师, 长期从事演化计算、机器学习、数据挖掘等方面的研究工作
E-mail: hubei_001@163.com