

SWO: 基于小世界效应的快速搜索算法

黄刚 李晋航 贾艳

(华中科技大学机械科学与工程学院工业与制造系统工程系 武汉 430074)

摘要 借鉴小世界网络理论中层次树模型和多分类标准建模的理论,设计了一种基于小世界效应的快速搜索算法 SWO。采用掩码规则将解空间构造为层次树网络模型,并提出采用相映射的空间与原解空间共同组成双分层标准的建模理论。SWO 算法通过对两种空间网络中长短邻居关系的查询访问,将实验信封推送到目的地,从而找到解空间中的最优值。实验证明,引入映射空间搜索机制可避免算法陷入局部最优,而长邻居关系的访问又加快了算法的收敛速度。通过与遗传算法(GA)、粒子群优化算法(PSO)和差分算法(DE)的对比,SWO 算法表现出较强的搜索能力和较高的搜索效率。

关键词 小世界优化算法,层次树网络模型,多分类标准建模,分布式搜索

中图分类号 TH181 **文献标识码** A

SWO: A Fast Search Algorithm Based on Small World Effect

HUANG Gang LI Jin-hang JIA Yan

(Department of Industrial and Manufacturing System Engineering, School of Mechanical Science and Engineering,

Huazhong University of Science and Technology, Wuhan 430074, China)

Abstract This paper designed a fast search algorithm called Small World Optimization(SWO) which was inspired by the hierarchical categorization tree model and multi categories method based on small world theory. The solution space can be divided into the hierarchical categorization tree model using mask rule in binary coding. Two bijective mapping solution space were adopted to establish multi categories method. SWO can find the optimal solution in the designed small world network by short and long distance neighbor relationship as pushing the mail to target. SWO was tested via a benchmark test functions in a simulation and the corresponding results show that two bijective mapping space can avoid the algorithm falling into early maturity and the long distance neighbor relationship can accelerate the convergence rate. Compared with the most popular optimization algorithm as genetic algorithm(GA), Particle Swarm Optimization (PSO) and Difference Algorithm(DE), the SWO algorithm is endowed with faster convergence ability to solve complex optimization problems.

Keywords Small world optimization, Hierarchical categorization tree model, Multi categories standard model, Distributed searching

小世界研究起源于1929年一位匈牙利作家 F. Karinthy 提出的“地球上的任何两个人都可以通过一条由5位联系人组成的链条联系起来”的论断^[1]。1967年, S. Milgram 通过著名的信件投递试验证实了小世界现象^[2], 并提出“六度分离”(Six Degrees of Separation)理论。Watts 和 Strogatz 对小世界效应的研究^[3,4], 使小世界现象逐渐被重视并迅速成为复杂系统及复杂性理论研究的热点。

小世界网络已被证明按照一定策略构成的网络拓扑具有分布式快速搜索特性^[5]。小世界效应用于优化算法的探索, 最初见于 Kleinberg^[6] 对局部搜索贪婪算法和 Walsh^[7] 对图论中搜索问题的研究。Kleinberg 证明了由规则网格按照一定策略构成的小世界拓扑具有实现快速搜索的性质。Walsh 指出大多数图拓扑都具有小世界结构特征, 并提出采用随机

与快速重置策略来提高算法的全局搜索效率。

受小世界效应的启发, 已有学者做了一些有益的尝试。Chen^[8] 指出, 调整不同传播概率参数会影响小世界快速搜索特性。穆华平^[9] 提出一种具有动态邻域结构的微粒群算法: 当邻域多样性小于给定阈值时, 构造小世界拓扑以小概率向网络随机增加长距离边, 形成动态领域结构。该策略能获得更好的收敛性能和收敛速度。杜海峰^[10] 基于 Kleinberg 的网格模型设计了短距离和长距离搜索算子, 提出了一种小世界算法。陈煜聪等^[11] 和袁明新等^[12] 随后分别对上述算法会在搜索后期停滞以及局域短连接的搜索效率低问题, 提出采用随机抛出和根据信息熵调节持信群体个体浓度的改进方法。杨新艳^[13] 借鉴 Kleinberg 模型提出了一种具有较快收敛速度和良好搜索性能的分散式搜索策略。

到稿日期: 2010-08-04 返修日期: 2010-11-11 本文受国家自然科学基金资助项目(50775089, 50825503)资助。

黄刚(1972-), 男, 博士, 副教授, 主要研究方向为制造执行系统、生产调度优化、智能优化算法, E-mail: huanggang@hust.edu.cn; 李晋航(1983-), 男, 博士生, 主要研究方向为智能优化算法研究及应用、云制造、物联网, E-mail: jinhangli@gmail.com(通信作者); 贾艳(1983-), 女, 博士生, 主要研究方向为生产作业仿真。

本文提出一种基于小世界效应的小世界优化 (Small World Optimization, SWO) 算法。

1 算法基本框架

1.1 基本思路

SWO 算法的基本思路是: 模拟 Milgram 的信封投递实验, 从解空间不同地方投递若干个信封到特定目标, 将每一个信封持有人 (或称节点) 作为一个候选解; 每个节点根据目标函数的评价值来判断与其有联系的节点中最接近目标点的节点, 然后将信封传给这样的节点; 每个信封在小世界网络中通过多次投递过程的循环, 就会向目标节点靠拢, 从而达到寻优的目的。

但是, 基于 Kleinberg 小世界网络的快速搜索模型^[5]存在两个方面的缺陷: 首先, 实现快速搜索的建网条件苛刻, 可扩展性不强。其次, X 的维度过高, 会使解空间建立超高维网络, 网络规模的增加使平均传播步数增加, 从而降低算法收敛速度。因此, 有必要引入一种新机制来克服这种局限性。

引入 Watts 的层次树模型^[14]可解决上述问题。Watts 层次树模型解释了真实的社会网络结构: 个体根据职业、地理位置或兴趣等聚集成一些比较小的群, 这些群又根据它们的共同特征聚集成规模更大的群。这样一层一层向上聚集, 最高的一层代表整个网络, 从而产生一个树状的层次结构。同时, Watts 认为个体聚集成群的标准通常不止一种, 一个完整的网络通常有多种分层的标准。在实际的社会网络中, 人们会根据各种各样的标准来判断两人之间的距离, 除地理位置这种重要的依据外, 还有职业、国别、受教育程度和兴趣爱好等。与 Kleinberg 网络模型相比, 层次模型平均传递步数与网络规模无关, 并且实现可扩展路由的条件更宽松。

基于上述理论, 可以将解空间通过编码机制建立节点之间的层次树结构, 来缩小网络中任意两个节点之间的平均距离, 进而缩小搜索空间; 可以通过引入双射空间, 借助于解空间邻域关系的变换来实现两种标准层次树模型的映射和转换, 进而避免在某一个空间中陷入局部最优。以下分别介绍其具体策略。

1.2 小世界的层次树结构

根据 Watts 的层次树模型特点, 将小世界中的节点按照邻域关系分成不同团体, 建立团体之间的层次关系。即小团体组成大团体, 大团体又可以组成更大的团体, 进而建立节点之间的层次树关系。在一个层次树结构中, 定义节点 i 和 j 最近共同分支与其相隔的层数为它们之间的距离 d_{ij} , 若节点 i 和 j 在同一个组中, 则它们构成短邻居关系, 且 $d_{ij} = 1$ 。若节点 i 和 j 不在同组中, 则向上找到的共同分支的层数越高, 它们之间的距离就越大, 如图 1 所示。

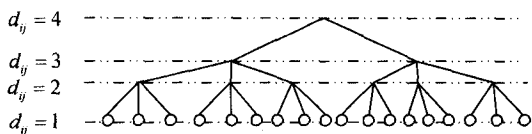


图 1 层次树结构示意图

按照 Watts 理论, 真实网络中处在不同组中的两个节点 i 和 j 之间也有可能为熟人关系, 但是两个节点的距离越大, 它们之间相互认识的概率越小。不同组中两节点相识的概率满足 $e^{-\alpha d_{ij}}$ 。通过相识概率连接的节点构成长邻居关系。其中, α 是相识指数。当 α 变大时, 节点趋于只与小组成员相识; 当 α 变小并趋于 0 时, 每个节点以相同概率相互连接, 就产生了

一个均匀随机图。 α 取值不同对分布式搜索的效率有影响。

通过引入层次关系, 减少了节点间的网络距离, 从而可以降低解空间的规模。

1.3 小世界中的空间转换

社会网络通常存在多重分类标准。设 h 为分类的标准数, 且 $h=1, 2, \dots, H$ 。这样, 小世界中每个节点的坐标就由一个 H 维的坐标向量 \vec{v}_i 来表示, v_i^h 表示节点 i 在第 h 个分类标准中的坐标。考虑到社会距离强调相似性而非差异性, 取 $d_{ij} = \min_h d_{ij}^h, h=1, 2, \dots, H$ 。

Watts 已证明, 几乎所有能够实现快速搜索的层次树模型参数都满足 $\alpha > 0, H > 1$ 。当 $H=2$ 时网络传递效率比 $H=1$ 时大大提高。但随着 H 的进一步增加, 传递效率反而降低^[14]。因此, 本文取 $H=2$, 即双重空间网络。为实现双重空间, 需要设计一对双射映射, 并根据解节点的邻域关系来建立双重空间层次树模型。双射一方面能够保证节点在双重空间中的唯一性, 另一方面的目的是尽量使解节点邻域关系被打乱, 实现同一个节点分别在两个空间中的邻居不同。借助于双射空间的邻域关系, 可在某一空间陷入局部最优时, 通过另一映射空间的搜索跳出局部极值。

1.4 算法流程

SWO 算法流程表述如下:

- 1) 设置迭代计数器 $c=0$ 。
- 2) 产生初始持信群体 $Q(t_0)$: 初始化时, 根据约束定义将解空间与编码建立映射关系, 通过随机选取机制得到每个信封的出发节点。
- 3) 查询邻居节点: 对每个持信节点抽取一定数量 n 的短邻居和长邻居进行查询。查询的邻居节点包括映射空间中的邻居节点和原始解空间的邻居节点。
- 4) 评价邻居节点: 计算邻居节点的评价函数, 并对其排序。
- 5) 投递信封: 选择比当前持信节点评价好的节点作为下一收信节点并选中进入持信群体, 将当前节点从持信群体中删除。选择策略有两种: 一是计算所有选中的邻居, 取评价价值最好的一个; 或者由于映射空间的邻居容易找到高质量的节点, 先从该空间邻居节点开始依次计算所有邻居, 找到一个比持信节点好的点就投递, 计算停止。
- 6) 判断是否满足终止条件: 如满足, 则结束搜索并打印结果, 否则继续;
- 7) 更新持信群体: 当所有信封都进行了投递操作, 信封所在的节点便组成了新的持信群体, 得到 $Q(t+1)$;
- 8) $c=c+1$, 返回步骤 3)。

SWO 需要控制的参数有编码长度 n 、小组成员数 mc 、不同组节点间相识概率指数 α 和测试信封数量 m 。 n 与问题规模和解空间分辨率有关, 取值太大会增加计算复杂度, 太小会遗漏更优的解。 mc 是层次树分组最小单位的节点和, 影响短邻居的数量, 它与分叉率 b 决定了层次树的网络结构。 α 是得到长邻居的重要参数, 一般取 0.4~0.6。 m 即持信群体大小, 一般取 10~40。

2 算法设计

2.1 小世界构建

持信群体是由当前所有持信封的节点组成的。每个节点对应一个问题的解。初始持信群体一般是随机产生的, 具体的产生方式依赖于编码方法。持信群体的大小依赖于计算机

的计算能力。为了建立层次树模型,本文编码方法采用二进制方式。假设 n 为编码长度,对于节点编码 $X = \{x_1, x_2, \dots, x_m\} \in R^m, x_i = \{b_{i1}, b_{i2}, \dots, b_{in}\}, i = 1, 2, \dots, m$, 初始持信群体产生方法如下:随机产生 $\xi_i \in U(0, 1]$, 使得

$$b_i = \begin{cases} 1, & \xi_i > 0.5 \\ 0, & \xi_i \leq 0.5 \end{cases} \quad (1)$$

重复产生 m 个节点编码,就得到一个初始化的持信群体。某些情况下,持信群体大小 m 可以变化,比如设定一种信封投递衰减率,使得有的信封因投递失败而退出持信群体,也可以新增信封来弥补持信群体数量的不足。

2.2 基于层次树模型的编码机制

借鉴互联网 ip 地址分组策略,将节点按照编码位(bit)来分组,通过掩码(mask)可以求得节点所属的组和组中的成员编号,也可以方便求出两个节点的层次距离,以及根据层次距离找出需求的节点。编码思想如下:

设节点编码 $X = \{x_1, x_2, \dots, x_m\} \in R^m, x_i = \{b_1, b_2, \dots, b_n\}, i = 1, 2, \dots, m$, 对应掩码为 $M = \{m_1, m_2, \dots, m_n\}$, 其中 $b_i, m_i \in \{0, 1\}, i = 1, 2, \dots, n$ 。

掩码为 1 的位代表组编码,为 0 的位代表小组中成员编码,设节点编码的组编码位数为 s, g 表示小组成员的编码位数,编码长度为 n ,该节点的掩码如下所述:

$$M = \{m_1^1, m_2^1, \dots, m_s^1, m_{s+1}^0, m_{s+2}^0, \dots, m_n^0\}, m_i^0 = 0, m_i^1 = 1, i = 1, 2, \dots, n$$

则组编码为

$$G_X = X \& M \quad (2)$$

成员编码为

$$I_x = G_X \wedge X \quad (3)$$

通过掩码的设定,解空间组成的网络可以构成一个类似计算机网络的层次树模型。组编码代表了该节点隶属的组号,成员编码代表了该节点在组中的唯一标识。组编码之间又构成了层次树关系,组的高位数代表该组的上层父代码。位数越高,代表层次树中的层次就越高。每个分叉点不代表实际节点。4 层层次树及其编码方式如图 2 所示,图中 XXX 表示取任意的 0 和 1。

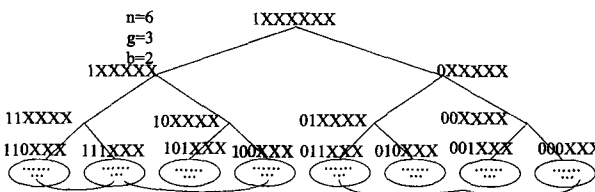


图 2 二进制编码构成层次树关系

对于多维解结构,两个节点 i 和 j 在 x_D 维的层次树距离为

$$d_{ij|x_D} = ||x_{iD} - x_{jD}|| = \Delta(G_{x_{iD}} \wedge G_{x_{jD}}) \quad (4)$$

$\Delta(x_{mn})$ 运算是求从左边位开始第一位不为 0 的位起到最右边位的位数和, $d_{ij|x_D} > 0$ 。两个节点 i 和 j 在整个解空间中的层次树距离为

$$d_{ij} = ||X_i - X_j|| = \min(d_{ij|x_D} | D = 1, \dots, m) \quad (5)$$

2.3 空间转换的实现

根据上述讨论,SWO 算法需建立双重空间,一套采用原始解空间,一套采用经过双射变换的空间。经过试验,格雷码辅助逆转变换(Gray and Reverse)映射空间可实现小世界快速搜索算法的要求。格雷码的任意相邻两码只有一位码元不相同,即相邻两代码的海明距离恒为 1^[15]。其几何意义在于,

沿格雷码的排列顺序可以走遍对应几何空间的所有点且走过的距离最短^[16]。

设二进制编码 $B = \beta_n \beta_{n-1} \dots \beta_1$ 对应的格雷码为 $G = \eta_n \eta_{n-1} \dots \eta_1, g(x)$ 是二进制转为格雷码的映射函数,则由二进制编码转格雷码的转换公式可表示为

$$g(B) = \begin{cases} \eta_n = \beta_n \\ \eta_i = \beta_{i+1} \wedge \beta_i, i = n-1, n-2, \dots, 1 \end{cases} \quad (6)$$

由格雷码转为二进制的转换公式可表示为

$$g^{-1}(B) = \begin{cases} \beta_n = \eta_n \\ \beta_i = \beta_{i+1} \wedge \eta_i, i = n-1, n-2, \dots, 1 \end{cases} \quad (7)$$

设 $r(x)$ 是将编码进行逆转的函数,且

$$r(B) = \beta_1 \beta_2 \dots \beta_n, B = \beta_n \beta_{n-1} \dots \beta_1 \quad (8)$$

则格雷码辅助逆转变换的映射函数为 $f(x) = g(x)r(x)$ 。

格雷码变换^[17]及其逆转变换都满足双射变换,因此采用 $f(x)$ 的空间映射是一一对应的。为简便起见,将二维空间经格雷辅助逆转变换后的空间称作格雷空间。图 3 显示了二维 $[0, 1]$ 空间及其格雷空间的映射关系,图 4 是格雷空间的矩形领域转换为原解空间的分布图。

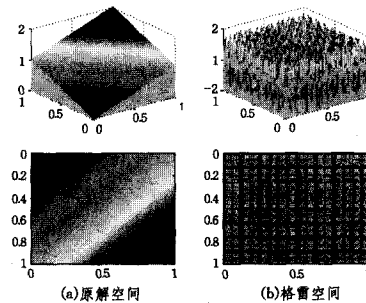


图 3 双重空间映射关系

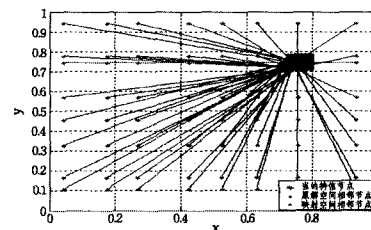


图 4 格雷空间某矩形邻域节点在原解空间的分布

可以发现,格雷变换改变了原解空间连续邻域的特征,使格雷空间中的邻域映射到原始空间后广泛分布在全空间范围,从而使节点中产生传播孤岛的可能性大大降低,同时提高了单节点搜索的全局性,有助于建立符合小世界快速搜索特性的网络。

2.4 分布式搜索算法

这个过程包括两个步骤:一是搜寻并评价自己的邻居节点,即查询算子;二是根据评价价值来选择合适的投递节点,即投递算子。

2.4.1 查询算子

(1) 短邻居查询

假设 X 的维数为 D , 小组成员数 $mc = (2^g - 1)^D$, 短邻居查询步骤如下:

步骤 1 设定要查询的邻居个数 $ns \leq mc$, 短邻居的最大地理距离 $sd \in \mathbb{Z}^+$;

步骤 2 随机生成 D 维符号向量 $\vec{sign} = s_1 s_2 \dots s_D, s_i \in \{+, -\}, i = 1, 2, \dots, D$;

步骤 3 根据下列伪代码(见图 5)得到一个邻居 X' , 其

中 $bit(1)$ 为最小距离单位;

// 查询一个短邻居:

$X' = X;$

Repeat 1; sd

$r = \text{randint}(1 \sim D);$ // 得到 1 到 D 之间的随机整数

$x_r' = x_r + s_r \cdot bit(1);$ // 每维移步单位 1, X 整体移位 sd

end;

output X'

图 5 查询一个短邻居流程

步骤 4 重复步骤 2 和步骤 3 ns 次, 得到规定数目的短邻居。

(2) 长邻居查询

长联系人的查询步骤如下:

步骤 1 采用轮盘赌的方式以 e^{-ad_i} 为概率选择 d_i 作为 X_i 要寻找的长联系人的距离。

步骤 2 随机选取 $x_j \in X_i$, 对其从左数 $n-g-d_i+1$ 位起至右末位进行反转 0-1 编码值变异操作。

步骤 3 为每一维值 $x_j (j=1, 2, \dots, D)$ 产生随机距离 $d_j \in \{1, 2, \dots, d_i\}$, 并对每个 x_j 从左数 $n-g-d_j+1$ 位起至右末位进行变异操作。

步骤 4 变异操作结束后得到的 X_i' 是 X_i 的长距离联系人。

2.4.2 投递算子

投递算子采用 Milgram 贪婪算法: 当前节点 i 把信封交给所有在原始解空间和映射解空间的短距离联系人和长距离人中认为最接近目标节点 t 的节点 j 。由于算法的目的是找到目标节点, 并不知道节点 i 的具体位置, 因此可以认为评价价值越高的节点越靠近目标节点。

2.5 解码

当对节点进行评价或者输出最终结果时, 需要解码。对于 D 维连续优化函数 $f(x)$, $x = (x_1, x_2, \dots, x_D)$, $x_i \in [u_i, v_i]$, $i=1, 2, \dots, D$, 设其二进制编码各维变量编码长度为 l_i , 则 x 构成的总长度为 $L = \sum_{i=1}^D l_i$ 的二进制编码串。 x_i 位段解码

函数 $\Gamma^i: \{0, 1\}^{l_i} \rightarrow [u_i, v_i]$ 的形式为

$$x_i = \Gamma^i(a_1^i, a_2^i, \dots, a_{l_i}^i) = u_i + \frac{v_i - u_i}{2^{l_i} - 1} \left(\sum_{j=1}^{l_i} a_j^i 2^{l_i - j} \right), \quad i=1, 2, \dots, D \quad (9)$$

式中, a_{l_i} 是 x_i 第 l_i 位的二进制位。通过上式运算可以求得各 x_i 的真实值, 从而得到 x 。

2.6 终止准则

终止准则可采用最大迭代次数的方法。当传递步数达到上限值, 算法终止。

由于小世界优化算法的特殊性, 终止策略还可以采用消息耗损率来表示。Watts 认为每个信封交给下个节点具有耗损率 p_e , 即每一步消息终止的概率为 p_e 。消息最终达到目标节点的概率 q 不低于设定值 r , 于是有

$$q = [(1 - p_e)^L] \geq r \quad (10)$$

最大的平均传递步数 L 可由以下不等式表示

$$L \leq \ln r / \ln(1 - p_e) \quad (11)$$

若取 $p_e \approx 0.25$, $r = 0.05$, 则 $L \approx 10.4$ 。参数值可以根据问题复杂性的实际情况设定。

3 实验分析

为了进行定量比较, 采用 8 个经典测试函数 $f_1 - f_8$ 进行实验分析, 分别是 Rosenbrock 函数^[18]、Camel 函数^[19]、Rastrigrin 函数^[19]、Shekel's Foxholes 函数^[20]、Schaffer's 函数^[20]、Schwefel's 函数^[21]、Shubert 函数^[22] 和 Hansen 函数^[22]。利用这 8 个测试函数, 在 CPU2.5GHz/1G 内存的计算机上用 matlab2009b 编程测试本算法 (SWO), 并与差分算法 (DE)、遗传算法 (GA)、粒子群优化算法 (PSO) 算法进行对比分析。

3.1 求解测试与结果分析

参数选择上, SWO 编码长度 $n=20$, 小组成员码长 $g=3$, 分叉率 $b=2$, 初始信封数 $m=10$, 长邻居连接概率 $\alpha=0.5$ 。算法终止条件为评价次数 $EFs < 1e5$, 若结果达到给定精度, 则算法提前终止。

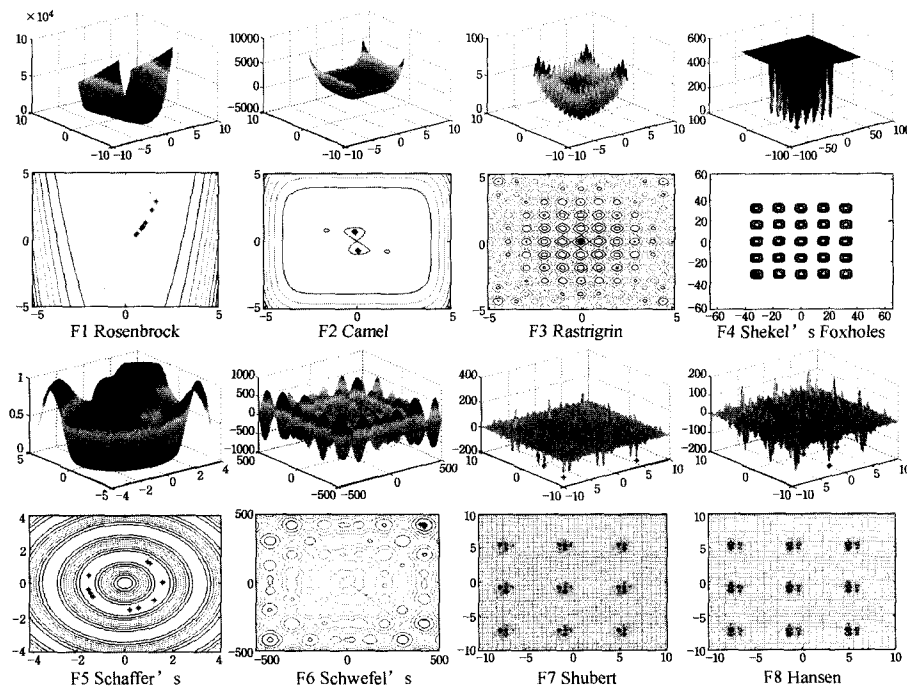


图 6 8 个经典测试函数 SWO 测试结果示意图

图6是SWO测试这8个经典测试函数的结果图。图中‘*’代表SWO达到求解精度时找到的最好解位置(f_s 求最大值,其余求最小值)。

实验发现,原解空间短邻居(Real-short)、原解空间长邻居(Real-long)、映射空间短邻居(Gray-short)、映射空间长邻居(Gray-long)对算法的贡献率不同。图7反映了在1万次评价计算中SWO的4种邻居关系改进的作用次数 τ 及其对总改进值贡献率 ν 的对比,其中,贡献率 ν =各项改进值(φ)/作用次数(τ)。可以看出,原解空间的邻居关系改进结果的作用次数远大于映射空间,且短联系人的改进次数又比长联系人多;但是,映射空间邻居关系对算法的贡献率却大于原解空间的邻居关系,且长邻居的收敛效率要大于短邻居的效率。

图8为算法迭代625次收敛的斜率。值越低,表明当前迭代时收敛的速度越快。由图可见,虽然SWO收敛速度的规律由高向低逐步递减,但总体保持了较快的收敛速度,迭代到后期也能保持一定的速率。

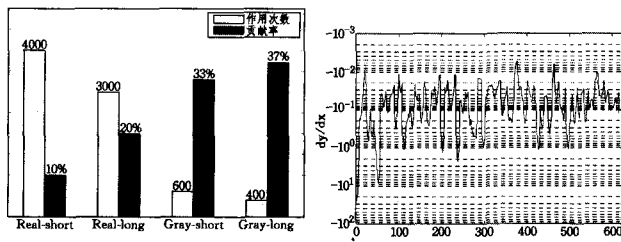


图7 SWO 4种邻居对算法的作用次数和贡献率对比

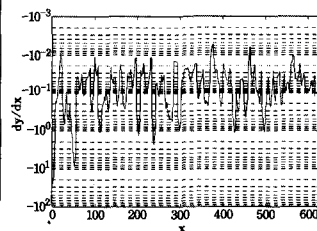


图8 SWO收敛斜率

3.2 对比实验分析

DE,GA,PSO初始群体大小为20,其余为默认参数。算法终止判断条件上,算法评价次数 $EFs < 1e5$,若达到测试精度范围,则算法自动终止。为了消除算法随机性对求解性能评价的影响,每个函数测试50次,结果见表1。其中,Object Attained为50次计算终止时达到计算精度的次数,EFs Mean为达到计算精度时的平均评价次数,EFs Std.为评价次数的标准差。

从表1可以看出,SWO在达到的计算精度和收敛速度两个方面比DE,PSO和GA表现更好。SWO的所有计算都能在给定的最大评价次数下收敛至给定的计算精度。在算法收敛速度上可以看出SWO在较少的评价次数下能实现快速收敛,仅在函数4的计算上稍逊于PSO。在算法稳定性上,从评价函数标准差的对比可以看出,DE和PSO体现出一定优势,但没有一种算法能够绝对领先。SWO在函数2的测试上胜于其他算法,而在其他函数测试中落后于DE和PSO。

总体看来,SWO求解优化函数上比其他标准进化算法(如DE,GA和PSO)在求解能力和收敛速度上都有一定优势。但是在算法鲁棒性上还需要进一步改进。产生这两种特点的原因主要是:前者,SWO建立了一套解空间的小世界网络拓扑,在这种网络下可以加速贪婪搜索算法的搜寻速度;后者,由于SWO节点信息并不共享,导致每个信封的实验是独立的。起点不同,并不是所有信封都能有幸在小世界网络中快速找到目标。这与Watts和Strogatz实验结果相似^[23],来自166个国家和地区的6万多名志愿者参与实验,结果只有384个志愿者的邮件抵达了目的地。

表1 函数测试实验结果统计

函数 Function	精度 Precision	Object Attained				EFs Mean			EFs Std.				
		SWO	DE	GA	PSO	DE	GA	PSO	SWO	DE	GA	PSO	
1	1.00E-03	50	30	10	25	4600	5600	8600	4900	32.5	23.43	57.9	12.3
2	1.00E-03	50	50	30	50	3500	5200	6840	4600	23.8	37.45	78.21	24.2
3	1.00E-02	50	50	45	45	1000	3000	5740	5210	154.92	104.51	340.78	145.3
4	1.00E-03	50	50	50	50	3700	3900	6530	3170	103.4	87.32	120.3	80.4
5	1.00E-04	50	43	21	36	390	1090	3070	1290	219.4	180.2	250.2	231.4
6	5.00E-01	50	50	50	50	4500	6700	8300	5700	20.3	25.34	23.43	13.4
7	1.00E-02	50	50	30	50	3000	5120	6400	6120	107.43	22.3	145.3	109.2
8	1.00E-01	50	31	20	50	5700	7900	8900	8250	33.51	12.4	34.5	230.3

结束语 SWO算法借鉴小世界网络可以实现快速搜索的“六度分离”的特性,将解空间按照Watts理论建立一套双重空间的层次树模型。通过分布式搜索算法在原解空间和映射解空间配合搜索,使当前解在其引领下向最优解靠拢。经过试验证明,本文算法在优化求解中是可行的,是一种智能优化算法。但是SWO也存在一些不足,比如信封之间相互独立,使得局部搜索带有盲目性,容易陷入局部极值。进一步将示参数的选择对算法收敛的影响,以及通过全局信息共享提高算法的鲁棒性,这将是SWO算法今后研究的重点。

参考文献

[1] Braun T. Hungarian priority in network theory[J]. Science, 2004,304(5678):1745 .
 [2] Milgram T S. An experimental study of the small-world problem [J]. Sociometry,1969,32
 [3] Watts D J, Strogatz S H. Collective dynamics of ‘small-world’ networks[J]. Nature,1998,393:440
 [4] Strogatz S H. Exploring complex networks[J]. Nature, 2001,

410(6825):268
 [5] Kleinberg J. The Small-world Phenomenon: An Algorithmic Perspective[C] // Proceedings of the Thirty-second Annual ACM. Ithaca, Cornell Computer Science Technical Report. 1999,10:99-1776
 [6] Kleinberg J. Navigation in a small world[J]. Nature,2000,406(6798):845
 [7] Walsh T. Search in a small world[C]//16th International Joint Conference on Artificial Intelligence. Stockholm, Sweden; Morgan Kaufmann Pub Inc,1999
 [8] Chen J Z, Liu W, Zhu J Y. Two-dimensional small-world networks; Navigation with local information[J]. Physical Review E,2006,73(5):6
 [9] 穆华平,曾建潮. 基于小世界模型动态演化邻域的微粒群算法 [J]. 系统仿真学报,2008,20(15):3940
 [10] 杜海峰,庄健,张进华,等. 用于函数优化的小世界优化算法[J]. 西安交通大学学报,2005,9(39):1011
 [11] 陈煜聪,杨斌,杜海峰,等. 一种具有跟踪替代特征的小世界算法 [J]. 西安交通大学学报,2007,11(41):1360

- [12] 袁明新,杜海峰,王孙安,等.一种基于种群熵的混沌小世界优化算法[J].西安交通大学学报,2008,9(42):1137
- [13] 杨新艳,王晓华.分散式小世界优化策略[J].苏州大学学报:工科版,2007,3(27):41
- [14] Watts D J, Dodds P S, Newman M E J. Identity and search in social networks[J]. Science, 2002, 296(5571): 1302
- [15] Sharma B D, Khanna R K. On m -ary Gray Codes[J]. Information and Control, 1978, 15: 31
- [16] 张公礼,许春香.混合进制格雷码[J].西北电讯工程学院学报, 1985, 4: 1
- [17] Conway J H, Sloane N J A, Wilks A R. Gray codes for reflection groups[J]. Graphs and Combinatorics, 1989, 5: 315-325
- [18] Price K V. Differential evolution: a fast and simple numerical optimizer [C]// 1996 Biennial Conference of the North American Fuzzy Information Processing Society. Piscataway, USA; IEEE Press, 1996: 524-527
- [19] Eberhart R C, Shi Y. Comparing inertia weights and constriction factors in particle swarm optimization [C]// Proceedings of the Evolutionary Computation. Piscataway, USA; IEEE Press, 2000: 84-88
- [20] 焦李成,杜海峰,刘芳,等.免疫优化:计算、学习与识别[M].北京:科学出版社,2006:402
- [21] Leung Y W, Wang Y. An orthogonal genetic algorithm with quantization for global numerical optimization [J]. Evolutionary Computation, 2001, 5(1): 41
- [22] Madsen K. Nonlinear approximation and engineering design [EB/OL]. Lyngby, Denmark; Information and Mathematical Modeling Institute. <http://www2.imm.dtu.dk/~km/GlobOpt/testex/testproblems.html>, 2006-12-21
- [23] Dodds P S, Muhamad R, Watts D J. An Experimental Study of Search in Global Social Networks[J]. Science, 2003, 301(5634): 827

(上接第 242 页)

对于回溯法,实际上就是将一个迷宫中的所有可行路径扫描一遍、所有“死”路扫描两遍(前进、后退一遍)并记录下所有“走过点”最后经过的方向,其时间复杂度为 $O(2 * n * 2n + n) = O(4n^2 + n)$ 。

显然, M-TdPN 算法具有较小的时间复杂度。

5 实验仿真

为验证 M-TdPN 算法的有效性,使用 HPsim 软件平台对本算法进行仿真实验。采用 Java 语言完成迷宫的自动生成,迷宫规模为 $m \times n$ 。图 5 所示分别为 $30 \times 20, 80 \times 65, 150 \times 120, 200 \times 100, 200 \times 200, 250 \times 350, 300 \times 300, 350 \times 300$ 8 种不同规模迷宫的实验结果。图中的实曲线表示经过 M-TdPN 算法求得最优路径的时间,图中虚线是以回溯法求得最优路径的时间。由图 5 可以看出,实曲线随着迷宫规模的扩大比虚曲线能在更短时间内快速接近最优解。随着迷宫复杂度的增加,回溯法的时间复杂度会趋向 $O(n^4)$,而 M-TdPN 算法保持了良好的鲁棒性。通过多次对随机产生的迷宫矩阵实验表明,算法 M-TdPN 总能找到最优解,其平均求解时间优于回溯法。

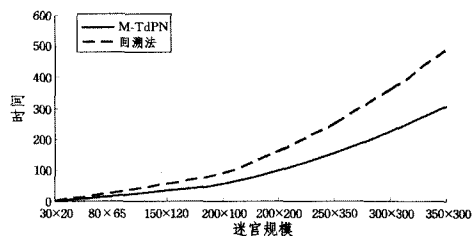


图 5 不同算法求解迷宫的实验结果

结束语 本文利用时延 Petri 网得到了一个新的、与传统思路完全不同的迷宫求解方法,并给出可实现的算法。本方法先对迷宫加以简化,接着逐步将其转换成时延 Petri 网,利用 Petri 网的并发特性求解通路。从实验结果来看,本文所提出的算法是有效的,对大规模复杂迷宫的求解效率比传统回溯法要高。在本文的后续工作中,还将进一步利用 Petri 网分析方法研究迷宫结构。

参考文献

- [1] 严蔚敏,吴伟民.数据结构(C语言版)[M].北京:清华大学出版社,2004:50-52
- [2] Dorigo M, Caro G D, Gambardella L M. Ant algorithms for discrete optimization [J]. Artificial Life, 1999, 5(3): 137-172
- [3] 胡小兵,黄席樾.蚁群算法在迷宫最优路径问题中的应用[J].计算机仿真,2005,22(4):114-116
- [4] 张公敬,徐熙君.蚁群算法求解迷宫最优路径[J].青岛大学学报:自然科学版,2008,21(1):61-65
- [5] Hachour O. Path planning of autonomous mobile robot [J]. International Journal of Systems Applications, Engineering and Development, 2008, 4(2): 178-190
- [6] 唐国新,陈雄,袁杨.基于改进遗传算法的机器人路径规划[J].计算机工程与设计,2007,28(18):4446-4449
- [7] 廖国勇,王广超.用遗传算法解迷宫问题[J].华东交通大学学报,2006,23(2):138-140
- [8] 黄猛,唐琳,胡世安,等.基于粗糙集理论与遗传算法的迷宫问题求解[J].现代电子技术,2009,311(24):144-150
- [9] Sutner K. Cellular automata and intermediate degrees[J]. Theoretical Computer Science, 2003, 296: 365-375
- [10] 赵学峰,张贵仓,王治和.基于细胞自动机的迷宫问题求解[J].西北师范大学学报:自然科学版,2006,42(3):29-31
- [11] Reisig W. Petri Nets; An Introduction [M]. Berlin, Heidelberg: Springer Verlag Press, 1985: 17-135
- [12] 袁崇义. Petri 网原理与应用 [M]. 北京:电子工业出版社,2005: 17-135
- [13] 吴哲辉. Petri 网导论 [M]. 北京:机械工业出版社,2006: 157-179
- [14] 卢开澄,卢华明.图论及其应用[M].北京:清华大学出版社,2002:3-40
- [15] van der Aalst W M P, van Hee K. Workflow Management: Models, Methods, and Systems [M]. MIT, Cambridge, MA; 2002: 208-227
- [16] 廖伟志,彭月英,古天龙.区间速率连续 Petri 网的模糊模型[J].计算机科学,2009,36(2):234-237
- [17] Murata T. Petri Nets: Properties, Analysis and Applications [J]. IEEE Trans on Software Eng, 1987, 77(4): 541-580