

基于引力搜索和分布估计的混合离散优化算法

蒋悦¹, 沈冬梅², 赵彦¹, 高尚策^{2*}

(1. 江苏信息职业技术学院 物联网工程系, 江苏 无锡 214153; 2. 东华大学 信息科学与技术学院, 上海 201620)

(* 通信作者电子邮箱 gaosc@dhru.edu.cn)

摘要:针对传统离散引力搜索算法(GSA)容易陷入局部最小解的问题,提出了一种引力搜索和分布估计的混合离散算法 GSEDA。通过有效地利用个体在引力搜索的历史统计信息,结合分布估计建立的概率分布模型,生成新的具有全局统计意义的优良解,继而更新搜索群体,使算法搜索更加平衡了空间的开发和探索能力,从而使得算法具有更强的跳出局部最优解的能力。仿真实验结果表明提出的新算法比传统算法具有更好的优化性能和鲁棒性。

关键词:引力搜索算法; 分布估计; 旅行商问题; NP-hard 问题; 局部最优解

中图分类号: TP391.4 **文献标志码:** A

Hybrid discrete optimization algorithm based on gravity search and estimation of distribution

JIANG Yue¹, SHEN Dongmei², ZHAO Yan¹, GAO Shangce^{2*}

(1. Department of Internet of Things, Jiangsu College of Information Technology, Wuxi Jiangsu 214153, China;

2. College of Information Sciences and Technology, Donghua University, Shanghai 201620, China)

Abstract: According to the problem of the traditional Gravitational Search Algorithm (GSA) such as falling into the local minimum point easily, a hybrid algorithm based on Estimation of Distribution (ED) and gravitational search (GSEDA) was proposed. By characterizing the distribution of current solutions found by GSA, ED was used to generate promising solutions based on the constructed probability matrix, thus guiding the search to new solution areas. The proposed GSEDA was able to balance the exploration and exploitation of the search, therefore possessing a better local optima jumping capacity. The experimental results based on the traveling salesman problem indicate that GSEDA performs better than traditional algorithms in terms of solution quality and robustness.

Key words: Gravitational Search Algorithm (GSA); Estimation of Distribution (ED); Traveling Salesman Problem (TSP); NP-hard problem; local optimal solution

0 引言

引力搜索算法(Gravitational Search Algorithm, GSA)是一种新型元启发式搜索算法^[1]。GSA 利用粒子群体对解空间进行搜索,群体中的所有个体的质量都对应着待解问题的适应度,其所处的空间位置对应着问题的解,个体间的作用力遵循万有引力定律。在算法初期,所有个体随机存在于搜索空间中的任意位置,随着时间的推移,个体由于受到其他个体的作用力影响而不断改变其自身位置,且质量越大的个体,对其他个体的引力越大,从而在宏观上造成所有个体都朝向质量最大的个体逐渐移动,并最终收敛到质量最大物体的位置^[1]。GSA 已经被证实了在连续^[1]和二值^[2]搜索空间上比粒子群算法^[3]以及有心力优化^[4]具有更加优良的搜索性能。目前,引力算法受到国内外学者的广泛重视,并成功应用于函数优化^[5]、聚类学习^[6]、图像处理^[7]和石油需求预测^[8]等实际复杂工程问题。

尽管 GSA 已经取得了一定的研究进展,但是它仍然受限

于传统的优化缺陷,如陷入局部最小化、进化早熟、搜索缓慢,缺乏有效的加速收敛机制^[9-11]等。近年来,众多学者分别从改善引力搜索的移动及选择算子,融合其他搜索机制等方面对引力算法提出了一些改进模型。徐遥等^[12]提出了权值化惯性质量的改进引力算法,并在基准函数测试上验证了算法的有效性。Sarafrati 等^[13]提出了分裂算子用于进一步探索当前最优个体的邻域范围。Shamsudin 等^[14]通过将位置更新规则修改成仅与个体的速度和方向相关,从而提高了算法的收敛速度。Saeidi-Khabisi 等^[15]提出了模糊引力搜索算法的模型,用来提高算法的进化效率和速度。Doraghinejad 等^[16]提出了基于天文现象的黑洞算子用于增强算法的空间探索能力,并在多模态优化问题上取得了成功。此外,引力算法与其他算法的融合,例如 K 均值聚类^[17]、免疫疫苗和记忆^[18]、支持向量机^[19]和混沌搜索^[20]等,均展现出鲁棒的搜索性能。作为一种随机搜索算法,GSA 并没有对其搜索到的空间位置的历史信息进行保存,而截止目前对 GSA 的改进算法也很少利用历史搜索信息对当前的搜索进行指导。与之相反,分布

收稿日期:2013-12-31;修回日期:2014-03-03。

基金项目:国家自然科学基金资助项目(61203325);上海市晨光计划项目(12CG35);教育部博士点新教师基金资助项目(20120075120004)。

作者简介:蒋悦(1983-),男,江苏宜兴人,助教,硕士,主要研究方向:计算智能; 沈冬梅(1988-),女,山东泰安人,硕士研究生,主要研究方向:进化算法; 赵彦(1981-),女,河南开封人,讲师,硕士,主要研究方向:软件技术; 高尚策(1983-),男,江苏泰州人,副教授,博士,主要研究方向:计算智能。

估计(Estimation of Distribution, ED)^[21]则能够对历史搜索信息进行维护和利用,并引导当前搜索朝着更有吸引力的解空间进行探索^[22]。

为了进一步提高引力算法的搜索性能,解决其易于陷入局部最优值的问题,本文提出一种基于引力搜索和分布估计得混合离散优化算法(Gravitational Search and Estimation of Distribution Algorithm, GSEDA)。有效地结合了引力搜索和分布估计的特点,利用引力搜索的快速探索能力使搜索能够迅速逼近局部最优解,再采用分布估计对搜索的历史信息实施建模和抽样,并引导搜索朝着更有吸引力的区域实施开发,进而使个体具备了良好的跳出局部最优解的能力。通过在典型的离散组合优化问题——旅行商问题(Travelling Salesman Problem, TSP)实例上的仿真实验,以及和离散引力搜索^[23]、离散粒子群优化(Discrete Particle Swarm Optimization, DPSO)^[24]和混合粒子群算法(Estimation of Distribution Particle Swarm Optimization, EDPSO)^[25]的性能比较,验证了GSEDA的有效性和鲁棒性。

1 分布估计

分布估计(ED)^[21,25-29]是一种新型的随机优化方法,通过统计学习的手段建立起解空间内个体分布的概率模型,进而由随机采样产生的新种群对原始种群进行进化更新。ED能够通过概率图模型对多变量之间的相关性进行统计建模,它没有传统的交叉、变异等遗传操作^[21],取而代之的是对上述统计概率模型的学习和采样,进而使得寻优过程具有了方向性,提高了优化计算的效率^[30]。

根据统计概率模型^[28]的不同,ED可以分为变量无关模型、双变量相关模型和多变量相关模型三类,其基本运行框架可以描述如下:

步骤1 随机产生初始种群 $X_{ini}(t)$ 。

步骤2 选择,从当前种群中根据某种机制选择出若干优秀个体组建成父代种群 $X_{parent}(t)$ 。

步骤3 建模,采用概率模型评估 $X_{parent}(t)$ 构建一个概率分布模型 $P(t)$ 。

步骤4 采样,依据概率分布模型采样产生新的种群 $X_{offspring}(t)$ 。

步骤5 判断是否满足终止条件,如满足则输出当前群体中的最优解;否则转至步骤2。

目前,ED已经在模式识别、软件测试、癌症分类、生物信息、工程优化、机器学习等众多领域取得了广泛且富有成效的应用^[27,30-32],且非常适合与其他类型的随机算法进行混杂搜索,如独立成分分析(Independent Component Analysis, ICA)与ED的混杂^[33]、差分进化(DE)与ED的混杂^[34]、粒子群优化(Particle Swarm Optimization, PSO)与ED的混杂^[25]、单纯形方法(Simplex Method, SM)与EA的混杂^[35]等。然而,就我们目前而知并没有引力搜索和ED的混合算法的相关报道。

2 基于分布估计的混合引力搜索算法

为了解决现有离散引力搜索算法容易陷入局部最优解的问题,本文提出将分布估计融合进引力搜索机制中的方法,通过将引力搜索到的解进行记忆存储在一个概率矩阵中,并建立出对所有优良历史候选解的概率分布统计模型,进而利用

采样产生新的候选解并加入到引力搜索的群体中,从而指导下一代过程中的引力搜索方向。本文提出的混合算法GSEDA将有效地结合引力搜索机制和分布估计的优点,是一种新的离散组合优化问题的解决方法。

2.1 离散引力搜索模型

假定 $X = \{x_1, x_2, \dots, x_i, \dots\}$ 表示组合优化问题的解的有限集合,其适应度函数为 $f: x \rightarrow \mathbf{R}$, 任何最小化问题的目标均可以表述为找出 $x_* \in X$ 使得 $x_* = \arg \min_{x \in X} f(x)$ 。若待解优化问题的解 $x_i \in X$ 可以被元素集 $[n] = \{1, 2, \dots, n\}$ 的一个线性顺序所表示,即 $\forall x_i \in X, x_i = (x_i^1 x_i^2 \dots x_i^n) \in \{1, 2, 3, \dots, n\}$, 显然, $|X| = n!$, 则上述向量中的任意一个整数值都编码了个体集中的相对顺序,并且被标记为个体的“排列”。为了使个体更加有利于操作算子的实现,本文采用双列形式来表达一个排列。

定义1 任意一个排序均是有限元素集 $[n]$ 到其自身的一个双射。排序可以采用关系记号法来表示,即元素的自然顺序被放置为上行,其变化后的元素被置为下行:例如 $\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 3 & 5 & 4 & 2 & 1 \end{pmatrix}$ 代表了元素集 $\{1, 2, 3, 4, 5\}$ 上的排列 s , 其中 $s(1) = 3, s(2) = 5, s(3) = 4, s(4) = 2, s(5) = 1$ 。

定义2 如果一种排列中存在一组唯一的 $[n]$ 中的 k 个元素 a_1, a_2, \dots, a_k 满足以下条件,则称此排列为一个回路:

$$s(a_i) = \begin{cases} a_{i+1}, & i = 1, 2, \dots, k-1 \\ a_1, & i = k \end{cases}$$

其中 $s(x) = x, \forall x \in [n], x \notin \{a_i\} (i = 1, 2, \dots, k)$ 。

2.2 搜索空间

让引力算法能够进行搜索,需要首先定义问题的解空间。令 $N: X \rightarrow 2^X$ 表示任意解 $x_i \in X$ 上的一个对应邻域函数,其邻域中的解 $x_j \in N(x_i) \subset X$ 可以通过应用一系列一维映射操作 $x_i \rightarrow x_j$ 来实现,用 Ψ 来表示这个操作,且假定其是可逆的,即 $x_j \in N(x_i) \Leftrightarrow x_i \in N(x_j)$ 。由此,通过定义邻域结构 $N(x_i)$ 以及由此而产生的距离函数 $h_{ij}(x_i, x_j)$ 能够将问题的搜索空间转化为度量空间。形式上,距离函数可以是任意的满足同一、对称及三角不等式原理的实值函数,从而其邻域结构可以表示为: $N(x_i) = \{x_j | x_j \in X, h_{ij}(x_i, x_j) \leq L\}$ 。在文献[36]中已经存在很多形式的距离函数,如汉明距离、边距离、偏差距离等。为了计算两个解之间的不同边的数目,通过定义恰当的数据结构,可以在 $O(n)$ 的时间复杂度内计算出其距离。

定义3 边距离度量函数 $h_{ij}(x_i, x_j)$ 被定义如下:

$$h_{ij}(x_i, x_j) = n - \sum_{i=0}^{n-1} \text{common}_{\text{edge}_k}$$

其中,

$$\text{common}_{\text{edge}_k} =$$

$$\begin{cases} 1, \exists l \in [0 \dots n-1] \text{ s.t. } e(x_{j1}, x_{j(l+1) \% n}) = e(x_{i(k+n-1) \% n}, x_{ik}) \\ 1, \exists l \in [0 \dots n-1] \text{ s.t. } e(x_{j1}, x_{j(l+1) \% n}) = e(x_{ik}, x_{i(k+1) \% n}) \\ 0, \text{其他} \end{cases}$$

显然, $h_{ij}(x_i, x_j)$ 的最大值为排列的长度 n 。

2.3 引力移动算子

在引力搜索算法中,个体移动算子 Ψ 定义了从位置 x_i 移动到其邻域中位置 $x_j \in N(x_i)$ 的操作,算法中的引力 F_{ij} 亦依赖于这个操作算子。

定义4 定义 Ψ 为基本移动算子, $\Psi = MO(d_1, d_2)$ 通过互换解的位置中两个分量的元素 d_1 和 d_2 , 同时满足 $X \rightarrow X$, $d_1, d_2 \in \{1, 2, \dots, n\}$, 且 $d_1 \neq d_2$ 。

举例1 将基本移动算子 $MO(1, 2)$ 应用于排列 $x_i = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 3 & 5 & 2 & 4 \end{pmatrix}$ 上, 移动后的位置 $x'_i = x_i \oplus MO(1, 2) = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 3 & 1 & 5 & 2 & 4 \end{pmatrix}$, 其中符号 \oplus 表示移动中的加法操作。类似地, 我们可以定义减法操作 \ominus 为: $x'_i \ominus x_i = MO(1, 2)$ 。显然, 这种基本移动算子其实是受进化算法中的变异算子的启发而设计而成, $MO(1, 2)$ 的移动操作相当于被操作个体中第一和第二个位置上的元素的互换。

定义5 定义移动序列 MS 为一个以上基本移动算子组合而成的移动序列: $MS = \{MO_1, MO_2, \dots, MO_n\}$, 其中 MO_i ($i = 1, 2, \dots, m$) 是基本移动算子。 MS 是这些基本算子的串联集合, 它们出现的顺序对 MS 有着本质的影响。通过移动序列, 我们可以将当前个体位置 x_i 移动至任意目标位置 x_j , 而移动序列的计算则是受进化算法中的部分映射交叉算子启发而得来, 见算法1:

步骤1 初始化 $d_1 = 1, k = 0, MS = \emptyset$;

步骤2 若 $d_1 \leq N$, 转步骤3; 否则结束算法并输出 MS ;

步骤3 若 $x_i^{d_1} \neq x_j^{d_1}$, 找出 d_2 使得 $x_i^{d_2} = x_j^{d_1}$, 并且 $d_1, d_2 \in \{1, 2, \dots, N\}$;

步骤4 设置 $MO(d_1, d_2)$ 且将其存储为 MS 的第 k 个元素;

步骤5 更新 $x_i = x_i \oplus MO(d_1, d_2), k++, d_1++,$ 转步骤2。

举例2 考虑一下两个解 $x_i = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 3 & 1 & 5 & 4 \end{pmatrix}$ 和 $x_j = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 2 & 3 & 4 & 5 \end{pmatrix}$ 。因为 $x_i^3 = x_j^1$, 第一个基本移动算子为交换 x_i 中位置1和3上的元素, 之后 x_i 便被移动至 $x_i = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 3 & 2 & 5 & 4 \end{pmatrix}$ 。重复上面的操作, 便可得到 $x_j = x_i \oplus MS$, 而 $MS = \{MO(1, 3), MO(2, 3), MO(4, 5)\}$ 。

2.4 算法描述

GSEDA 通过将问题的候选解看成搜索空间中的个体, 其质量为解的适应度, 随着个体的移动 (即基本移动算子及其序列的作用), 整个搜索群体朝着局部最优解收敛, 同时利用分布估计对引力搜索产生的优良解进行成概率模型的建立, 进而采样产生的新解加入到个体群体中, 重新进行轮盘选择。其基本思想如图1所示。

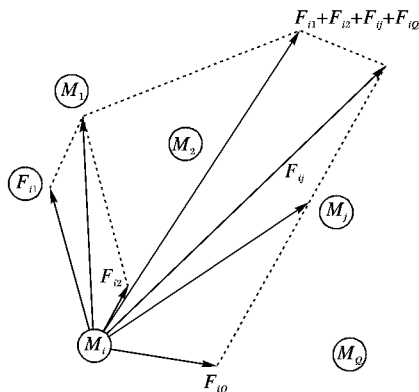


图1 基于分布估计的引力搜索算法的基本原理示意图

GSEDA 的群体中有 Q 个个体 $\{x_1, x_2, \dots, x_Q\}$, 每个个体

的质量为 M_i , 其在搜索空间中的位置是 $x_i = \begin{pmatrix} 1 & 2 & 3 & \dots & N \\ x_i^1 & x_i^2 & x_i^3 & \dots & x_i^N \end{pmatrix}$, 个体之间的引力 $F_{ij} = \sum_{j=1, j \neq i}^Q F_{ij}$, 其中: F_{ij} 是 x_j 作用在个体 x_i 上的分力, F_i 是其合力。根据万有引力定理, 在 t 时刻有:

$$F_{ij}(t) = G(t) \frac{M_i(t)M_j(t)}{(h(x_i, x_j))^2} (x_j(t) \ominus x_i(t)) \quad (1)$$

为了标准化个体的质量 $M_i(t)$, 将个体的适应度 $f_i(t)$ 通过以下方式转变为个体的质量:

$$M_i(t) = \frac{best(t) - f_i(t)}{best(t) - worst(t)}; \quad \begin{aligned} best(t) &= \min_{k \in \{1, 2, \dots, Q\}} f_k(t), \\ worst(t) &= \max_{k \in \{1, 2, \dots, Q\}} f_k(t) \end{aligned} \quad (2)$$

式(1)中, $x_j(t) \ominus x_i(t)$ 将根据定义5产生一个移动序列 MS_{ij} , 而 $G(t)$ 是控制移动序列执行概率的参数。

为了寻求问题的最优解, 候选解位置的更新依赖于个体的速度及其加速度, 本文将个体的初始速度均设为0, 且其更新不依赖于之前的速度, 只跟当前的加速度相关, 即:

$$\begin{aligned} v_i(t) &= a_i = \frac{1}{M_i} F_i = \frac{1}{M_i} \sum_{j=1, j \neq i}^Q F_{ij} = \\ &= \left(\frac{1}{M_i} F_{i1} \right) \oplus \left(\frac{1}{M_i} F_{i2} \right) \oplus \dots \oplus \left(\frac{1}{M_i} F_{iQ} \right) = \\ &= MS_1 \oplus MS_2 \oplus \dots \oplus MS_Q = \\ &= \{MO_{11}, MO_{12}, \dots, MO_{1n_1}, MO_{22}, \dots, MO_{Qn_Q}\} \end{aligned} \quad (3)$$

需要注意的是, 在式(1)和(2)中, 引力前面的系数均表示其执行的概率。

GSEDA 采用了单变量边缘分布模型 (Univariate Marginal Distribution, UMD)^[21] 进行分布估计的建模。首先选择引力搜索到的候选解, 统计每个解中从城市 i 直接访问城市 j 的移动边的数目, 形成一个概率分布概率矩阵 $[p_{ij}]$ ($i = 1, 2, \dots, n; j = 1, 2, \dots, n$)。即若被选择个体解中的 $s(i) = j$, 则 $p_{ij} = p_{ij} + 1$, 因此该分布概率矩阵反映了历史搜索解的一个分布情况。在每一次迭代中, 按照群体增量学习规则 (Population Based Increased Learning, PBIL)^[22] 更新分布概率矩阵:

$$p_{ij}(t+1) = (1 - \lambda)p_{ij}(t) + \lambda \left(\sum_{k=1}^Q p_{kj} \right) / Q \quad (4)$$

其中 $\lambda \in [0, 1]$ 是学习率。按照 PBIL 规则, ED 根据分布概率矩阵采用生产新的候选解。需要注意的是, 在遍历 $[p_{ij}]$ 生成解时, 如果生成部分非法解, 将采取贪婪搜索法对其进行修正^[21, 37]。

GSEDA 的整个算法流程见算法2:

步骤1 在搜索空间中随机初始化所有个体的位置 x_k ($k = 1, 2, \dots, Q$), 并计算 f_k , 令 $i = 1$;

步骤2 若 $i \leq Q$, 转步骤3, 否则转步骤9;

步骤3 计算所有个体的质量 M_k ($k = 1, 2, \dots, Q$);

步骤4 若个体 $j(j \neq i)$, 计算边距离 $h(x_i, x_j)$ 及引力执行

概率 $G(t) \frac{M_i(t)M_j(t)}{(h(x_i, x_j))^2}$;

步骤5 根据算法1计算移动序列 MS_{ij} ;

步骤6 计算速度 $v_i(t) = \{MO_{11}, MO_{12}, \dots, MO_{Qn_Q}\}$;

步骤7 更新个体位置 $x_i(t+1) = x_i(t) \oplus v_i(t)$;

步骤8 $G(t+1) = \alpha G(t)$;

步骤9 $i++$, 转步骤2;

步骤10 利用ED对当前群体进行分布估计,并根据式(4)更新概率分布矩阵 $[p_{ij}]$;

步骤11 采用产生新的候选解 M ,加入到原始引力搜索群体中,采取轮盘选择机制,重新产生 Q 个新的个体种群;

步骤12 如算法终止条件满足,输出当前最优解,否则转步骤2。

综上所述,GSEDA具有以下特征:其一,通过利用边距离定义搜索空间及离散操作算子,GSEDA能够处理离散组合优化问题;其二,利用进化策略的思想及基本算子实现了个体在解空间中位置的移动,使引力搜索能够有效地向局部最优解移动;其三,利用ED对搜索历史进行分布估计并采样,进而指导群体移向全局最优解。

3 应用验证

3.1 旅行商问题

为了验证提出算法的有效性,本文将GSEDA应用在传统的旅行商问题(Traveling Salesman Problem, TSP)上。TSP是一种NP-hard组合优化问题,许多现实应用问题(如VLSI设计问题,X-射线晶体学等)最终均可以转化为变形的旅行商问题。数学形式上,旅行商问题是无向图 $G = (N, E)$ 上的一个组合优化问题,其中 $N = 1, 2, \dots, n, E = 1, 2, \dots, m$,且任意两个顶点 i 和 j 之间的连边上都有附带值 c_{ij} ,代表旅行商从顶点 i 移动到顶点 j 上的花费。旅行商的目的在于找出无向图 G 上的一个最小长度的Hamiltonian回路,其长度是回路中所有附带值的总和。令个体的位置为 $x_i = \begin{pmatrix} 1 & 2 & 3 & \dots & N \\ x_i^1 & x_i^2 & x_i^3 & \dots & x_i^N \end{pmatrix}$,回路长度可以用式(3)计算:

$$D(x_i) = \sum_{k=1}^{N-1} c_{x_i^k x_i^{k+1}} + c_{x_i^1 x_i^N}$$

本文只讨论对称旅行商问题,即对任意的顶点 i 和 $j, c_{ij} = c_{ji}$ 。

3.2 实验仿真

在实验中,本文采用了TSP标准测试集中的5个典型问题,分别为eil51、st70、eil101、pr136和kroA200,并将算法的终止条件设定为其最大迭代次数达到 T_{\max} 。硬件仿真环境是:Windows 7操作系统,Intel Core i3 CPU 2.40 GHz, 2 GB内存,Matlab R2009a编译软件。

首先,本文研究了算法对于参数的敏感性问题。在GSEDA中存在多个用户自定义参数。为了测试不同参数取值及算法本身随机性对搜索性能的影响,各算法独立运行30次,且采用受限的标准参数,其取值如下: $Q = 50, T_{\max} = 1000, \lambda = 0.8, G(0) = n^2, \alpha = 0.99$ 。

群体规模大小 Q 和最大迭代次数 T_{\max} 是决定算法的搜索能力和时间复杂度的重要参数。表1给出了不同 Q 和 T_{\max} 下的算法在实例pr136上的性能比较,其中PDM(Percentage of Distance Mean)表示解平均值的百分比,PDB(Percentage of Distance Best)表示解最优值的百分比, T 表示算法运行的时间。

$$PDM = (D_m - D_{opt}) / D_{opt} \times 100\%$$

$$PDB = (D_b - D_{opt}) / D_{opt} \times 100\%$$

且 D_m, D_b, D_{opt} 分别代表算法获取的解的平均值、最优值以及

已知的全局最优解。通过表1可以看出, Q 或者 T_{\max} 越大,算法的搜索能力越强,但所需时间复杂度也越高。

表1 群体规模、最大迭代次数及学习率的敏感性分析

参数	取值	PDM/%	PDB/%	T/s
Q	1	7.1	8.5	0.18
	10	5.3	2.7	2.05
	50	1.5	0	11.32
	100	1.8	0.1	21.75
	200	1.8	0	45.54
T_{\max}	500	5.6	7.5	5.78
	1000	1.5	0	11.32
	2000	1.5	0	23.01
	5000	1.1	0	57.35
λ	0	15.2	7.4	9.50
	0.3	5.5	4.8	11.32
	0.5	4.9	3.2	11.32
	0.7	2.1	0	11.32
	0.8	1.5	0	11.32
	0.9	1.4	0.2	11.32

仔细比较可以发现:当 Q 的值小于50时,算法的性能会随着 Q 的变大而快速变优;相反,当其大于50时,算法性能会有一定的退化现象,即 Q 的快速增大并不能使性能迅速提高,其原因在于群体的多样性会随着移动算子的作用快速降低,导致过大的群体规模并不能始终保持其多样性。

另一方面, T_{\max} 取值为1000时是保持算法性能和时间复杂度平衡的最优选择。由表1可以看出, T_{\max} 大于1000后,算法性能的提升并不明显,原因在于此时算法已经基本陷入了局部最小。

参数 λ 决定了PBIL对历史候选解的学习能力,因而是对全局范围内的探索能力的一种控制。学习率 λ 比较小时,算法对EDA的依赖程度越小,当 $\lambda = 0$ 时表示只有引力搜索机制在发挥作用。随着学习率 λ 取值的增长,全局探索的能力逐渐得到加强,当 $\lambda = 0.8$ 时,对搜索空间的开发和探索取得了一定的平衡,此时GSEDA的搜索性能最强。

表2 引力参数初值和引力参数退化系数的敏感性分析

参数	取值	PDM/%	PDB/%	T/s
$G(0)$	n	1.5	0.1	11.31
	n^2	1.5	0.0	11.32
	n^3	1.6	0.0	11.32
α	0.5	1.9	0.8	11.31
	0.8	1.6	0.0	11.32
	0.9	1.8	0.3	11.32
	0.99	1.5	0.0	11.32
	0.999	1.6	0.1	11.32

文献[1]中作者申明参数 $G(t)$ 对算法在求解连续型问题时的结果影响非常大(指数级)^[1],然而在GSEDA中,通过表2的结果却发现其对离散型问题的影响却非常有限。本文分别设定 $G(0) = \{N, N^2, N^3\}, \alpha = \{0.5, 0.8, 0.9, 0.99, 0.999\}$,并在实例pr136上进行了交叉测试。实验结果表明,引力参数初值 $G(0)$ 对算法所搜索到的最优解几乎没有什么影响,而引力参数退化系数对解的影响(平均值PDM从1.5%

变化到1.9%)也较为有限(约26%)。其原因在于 $G(t)$ 只是控制移动序列执行概率的参数,由于算法中移动算子本身对最优解的探索能力并不强(其探索能力主要依赖于群体间的协同开发机制);此外,分别估计模型的融合也减弱了算法对引力搜索特有参数的影响。实验中 $G(t)$ 在 $G(0) = n^2$, $\alpha = 0.99$ 时取得了较好的搜索性能。

其次,为了证明算法的性能及鲁棒性,GSEDA和离散引力搜索^[23]、离散粒子群优化(DPSO)^[24]及混合粒子群算法(EDPSO)^[25]进行了比较。需要强调的是,尽管许多依赖于问题的高效的算子,如LK或者是混沌变量、领域拓扑结构等的使用可以提升算法的搜索性能,本文只讨论最基本的算法模

型和算子,即GSEDA可以被所有基于排列结构的组合优化问题所采用,具有泛化的算法结构和机制(problem-independent)。

表3总结了四种算法基于30次实验的仿真结果。表中的记录数据分别是旅行商实例的名称、城市规模大小、已知最优解,以及四种算法各自的最优、最差及平均解的质量(与已知最优解的偏差百分比)。由表3可以发现GSEDA均能够找到五种问题实例的最优解,且具有比较小的最差偏差值和较好的平均值,这些结果均表明GSEDA比其他三种算法具有更好的搜索性能和鲁棒性。此外,在图2中还给出协同进化引力搜索算法针对实例eil51、eil101、kroA200找出的最优解。

表3 GSEDA算法与其他算法之间的性能比较

TSP 实例	已知最优解	规模大小	最优/%				最差/%				平均/%			
			DGSA	DPSO	EDPSO	GSEDA	DGSA	DPSO	EDPSO	GSEDA	DGSA	DPSO	EDPSO	GSEDA
eik51	426	51	0.01	0.00	0.00	0.00	0.03	0.01	0.00	0.00	0.02	0.01	0.00	0.00
st70	675	70	0.02	0.01	0.00	0.00	0.03	0.01	0.01	0.00	0.03	0.01	0.01	0.00
eil101	629	101	0.02	0.04	0.01	0.00	0.05	0.04	0.01	0.00	0.04	0.04	0.01	0.00
pr136	96 772	136	0.04	0.08	0.01	0.00	0.04	0.09	0.01	0.01	0.04	0.09	0.01	0.00
kroA200	29 368	200	0.11	0.16	0.01	0.00	0.11	0.18	0.02	0.01	0.11	0.17	0.02	0.01

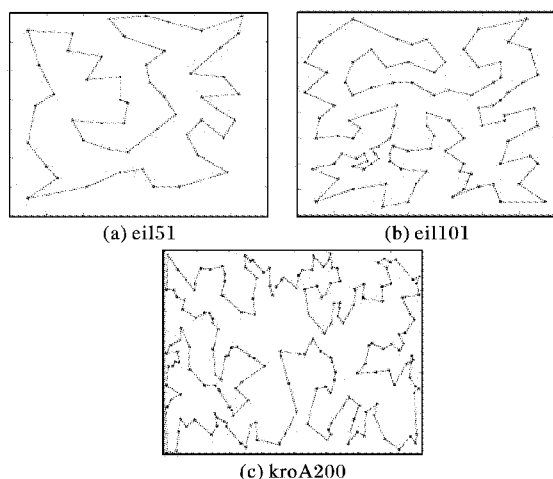


图2 GSEDA解决eil51、eil101、kroA200分别找到的最优解

4 结语

本文面向基于排列的离散组合优化问题的应用,通过将分布估计的概率估计和采样引进到引力搜索中,构造了以及分布估计的引力搜索算法(GSEDA)。分别定义了GSEDA中的搜索空间、移动算子及序列,利用单变量边缘分布模型对引力搜索的历史候选解进行的分布估计,建立了概率模型,并采样生产新解对原始种群进行更新。通过在传统旅行商问题上的应用以及与传统离散引力搜索、离散粒子群算法及离散混合分布估计粒子群算法的比较结果发现,GSEDA具有更好的搜索性能和鲁棒性。为了进一步解决大规模的问题实例,开发高效的移动算子是接下来的研究工作。

参考文献:

- [1] RASHEDI E, NEZAMABADI-POUR H, SARYAZDI S. GSA: A gravitational search algorithm[J]. Information Science, 2009, 179(13): 2232–2248.
- [2] RASHIDI E, NEZAMABADI-POUR H, SARYAZDI S. BGSA: Binary gravitational search algorithm[J]. Natural Computing, 2010, 3(9): 727–745.

- [3] KAMEYAMA K. Particle swarm optimization – A survey[J]. IEICE Transactions on Information and Systems, 2009, 7(92): 1354–1361.
- [4] FORMATO A. Central force optimization: A new nature inspired computational framework for multidimensional search and optimization[J]. Studies in Computational Intelligence, 2008, 129: 221–238.
- [5] JIN L, LI J, WEI P, et al. Maximal gravitation optimization algorithm for function optimization[J]. Pattern Recognition and Artificial Intelligence, 2010, 23(5): 653–662. (金林鹏, 李均利, 魏平, 等. 用于函数优化的最大引力优化算法[J]. 模式识别与人工智能, 2010, 23(5): 653–662.)
- [6] LI C, ZHOU J, FU B, et al. T-S fuzzy model identification with gravitational search based hyper-plane clustering algorithm[J]. IEEE Transactions on Fuzzy Systems, 2011, 19(6): 1–12.
- [7] LOPEZ-MOLINA C, BUSTINCE H, FERNANDEZ J, et al. A gravitational approach to edge detection based on triangular norms[J]. Pattern Recognition, 2010, 43(11): 3730–3741.
- [8] BEHRANG M, ASSAREH E, GHALAMBAZ M, et al. Forecasting future oil demand in Iran using GSA (Gravitational Search Algorithm)[J]. Energy, 2011, 36(9): 5649–5654.
- [9] LI P, DUAN H. Gravitation search algorithm based on improved UAV route planning[J]. Chinese Science: Technology and Science, 2012, 42(10): 1130–1136. (李沛, 段海滨. 基于改进万有引力搜索算法的无人机航路规划[J]. 中国科学: 技术科学, 2012, 42(10): 1130–1136.)
- [10] GANESAN T, ELAMVAZUTHI I, SHAARI K, et al. Swarm intelligence and gravitational search algorithm for multi-objective optimization of synthesis gas production[J]. Applied Energy, 2013, 103: 368–374.
- [11] RADU-CODRUT D, RADU-EMIL P, EMIL P, et al. Gravitational search algorithm-based design of fuzzy control system with a reduced parametric sensitivity[J]. Information Science, 2013, 247: 154–173.
- [12] XU Y, WANG S. Improved gravitational search algorithm[J]. Computer Engineering and Applications, 2011, 47(35): 188–192. (徐遥, 王士同. 引力搜索算法的改进[J]. 计算机工程与

- 应用, 2011, 47(35): 188 – 192.)
- [13] SARAFRAZI S, NEZAMABADI-POUR H, SARYAZDI S. Disruption: A new operator in gravitational search algorithm [J]. *Scientia Iranica D*, 2011, 18(3): 539 – 548.
- [14] SHAMSUDIN H, IRAWAN A, IBRAHIM Z, *et al.* A fast discrete gravitational search algorithm [C]// *Proceedings of the 4th International Conference on Computational Intelligence, Modelling and Simulation*. Washington, DC: IEEE Computer Society, 2012: 24 – 28.
- [15] SAEIDI-KHABISI F, RASHEDI E. Fuzzy gravitational search algorithm [C]// *Proceedings of the 2nd International eConference on Computer and Knowledge Engineering*. Piscataway: IEEE, 2012: 156 – 160.
- [16] DORAGHINEJAD M, NEZAMABADI-POUR H, SADEGHIAN A, *et al.* A hybrid algorithm based on gravitational search algorithm for unimodal optimization [C]// *Proceedings of the 2nd International eConference on Computer and Knowledge Engineering*. Piscataway: IEEE, 2012: 129 – 132.
- [17] YIN M, HU Y, YANG F, *et al.* A novel hybrid K-harmonic means and gravitational search algorithm approach for clustering [J]. *Expert Systems with Applications*, 2011, 38(8): 9319 – 9324.
- [18] ZHANG Y, WU L, ZHANG Y, *et al.* Immune gravitation inspired optimization algorithm [C]// *Proceedings of the 7th International Conference on Advanced Intelligent Computing*. Berlin: Springer-Verlag, 2012: 178 – 185.
- [19] SARAFRAZI S, NEZAMABADI-POUR H. Facing the classification of binary problems with a GSA-SVM hybrid system [J]. *Mathematical and Computer Modelling*, 2013, 57(1): 270 – 278.
- [20] LI C, ZHOU J, XIAO J, *et al.* Parameters identification of chaotic system by chaotic gravitational search algorithm [J]. *Chaos, Solutions and Fractals*, 2012, 45(4): 539 – 547.
- [21] HAUSCHILD M, PELIKAN M. An introduction and survey of estimation of distribution algorithms [J]. *Swarm and Evolutionary Computation*, 2011, 1(3): 111 – 128.
- [22] CREPINSEK M, LIU S, MERNIK M. Exploration and exploitation in evolutionary algorithms: a survey [J]. *ACM Computing Surveys*, 2013, 45(3): 1 – 35.
- [23] GAO S, TANG Z. Solving combinatorial optimization problems by a discrete gravitational search algorithm [J]. *ICIC Express Letters*, 2011, 5(7): 2119 – 2127.
- [24] del VALLE Y, VENAYAGAMOORTHY G, MOHAGHECHI S, *et al.* Particle swarm optimization: basic concepts, variants and applications in power systems [J]. *IEEE Transactions on Evolutionary Computation*, 2008, 12(2): 171 – 195.
- [25] ZHOU Y, WANG J, YIN J. An optimization algorithm based on estimation of distribution of discrete particle swarm [J]. *Acta Electronica Sinica*, 2008, 36(6): 1242 – 1248. (周雅兰, 王甲海, 印鉴. 一种基于分布估计的离散粒子群优化算法 [J]. *电子学报*, 2008, 36(6): 1242 – 1248.)
- [26] ZHANG Q, MUHLENBEIN H. On the convergence of a class of estimation of distribution algorithms [J]. *IEEE Transactions on Evolutionary Computation*, 2004, 8(2): 127 – 136.
- [27] ZHOU S, SUN Z. Estimation of distribution algorithms summary [J]. *Acta Automatica Sinica*, 2007, 32(2): 113 – 124. (周树德, 孙增圻. 分布估计算法综述 [J]. *自动化学报*, 2007, 32(2): 113 – 124.)
- [28] WANG X, CHENG Y, HAO M. Estimation of distribution algorithm based on bacterial foraging and its application in predictive control [J]. *Acta Electronica Sinica*, 2010, 38(2): 333 – 339. (王雪松, 程玉虎, 郝名林. 基于细菌觅食行为的分布估计算法在预测控制中的应用 [J]. *电子学报*, 2010, 38(2): 333 – 339.)
- [29] IVVAN VALDEZ S, HERNANDEZ A, BOTELLO S. A Boltzmann based estimation of distribution algorithm [J]. *Information Science*, 2013, 236(1): 126 – 137.
- [30] SIMONCINI D, ZHANG K. Efficient sampling in fragment-based protein structure prediction using an estimation of distribution algorithm [J]. *PLoS One*, 2013, 8(7): e689547.
- [31] SHIM V, TAN K, CHIA J, *et al.* Multi-objective optimization with estimation of distribution algorithm in a noisy environment [J]. *Evolutionary Computation*, 2013, 21(1): 149 – 177.
- [32] WANG L, FANG C, MU C, *et al.* A Pareto-archived estimation-of-distribution algorithm for multiobjective resource-constrained project scheduling problem [J]. *IEEE Transactions on Engineering Management*, 2013, 60(3): 617 – 626.
- [33] CHO D, ZHANG B. Evolutionary continuous optimization by distribution estimation with variational Bayesian independent component analyzers mixture model [C]// *Parallel Problem Solving from Nature 2004*, LNCS 3242. Berlin: Springer-Verlag, 2004: 212 – 221.
- [34] SUN J, ZHANG Q, TSANG E. DE/EDA: a new evolutionary algorithm for global optimization [J]. *Information Science*, 2005, 169(3/4): 249 – 262.
- [35] REN A, WANG Y, JIA F. A hybrid estimation of distribution algorithm and Nelder-Mead simplex method for solving a class of non-linear programming problems [J]. *Journal of Applied Mathematics*, 2013: No. 378568.
- [36] RONALD S. More distance functions for order - based encodings [C]// *Proceedings of the 1998 IEEE International Conference on Evolutionary Computation*. Piscataway: IEEE, 1998: 558 – 563.
- [37] LARRANAGE P, LOZANO J. Estimation of distribution algorithms: a new tool for evolutionary computation [M]. Hingham: Kluwer Academic Publishers, 2002.

(上接第 2073 页)

- [8] CUI M, SUN B. RWS-based floating point representation genetic algorithm [J]. *Journal of Convergence Information Technology*, 2012, 7(17): 459 – 467.
- [9] MARIANI V X, LUVIZOTTO G J, GUERRA F A. A hybrid shuffled complex evolution approach based on differential evolution for unconstrained optimization [J]. *Applied Mathematics and Computation*, 2011, 217(15): 5822 – 5829.
- [10] YOON Y, KIM Y-H, MORAGLIO S, *et al.* A theoretical and empirical study on unbiased boundary-extended crossover for real-valued representation [J]. *Information Sciences*, 2012, 183(1): 48 – 65.
- [11] CHANG D, ZHAO Y, ZHENG C, *et al.* A genetic clustering algorithm using a message-based similarity measure [J]. *Expert Systems with Applications*, 2012, 39(2): 2194 – 2202.
- [12] TANG P-H, TSENG M-H. Adaptive directed mutation for real-coded genetic algorithms [J]. *Applied Soft Computing*, 2013, 13(1): 600 – 614.
- [13] THAKUR M. A new genetic algorithm for global optimization of multimodal continuous functions [J]. *Journal of Computational Science*, 2013, 5(2): 298 – 311.