一种基于分布估计的离散粒子群优化算法

周雅兰, 王甲海, 印 鉴
(中山大学计算机科学系, 广东广州 510275)

摘 要: 本文提出了一种基于分布估计的离散粒子群优化算法. 提出的新算法突破了传统粒子群速度 - 位移搜索模型的局限, 且种群中的每个粒子具有更全面的学习能力, 从而能够有效地解决组合优化问题. 仿真实验结果表明提出的新算法的性能优于现有的其它几种离散粒子群优化算法.

关键词: 离散粒子群算法; 分布估计; 二分图问题; 组合优化问题
中图分类号: TP18 **文献标识码**: A **文章编号**: 0372-2112 (2008) 06-1242-07

A Discrete Particle Swarm Optimization Algorithm Based on Estimation of Distribution

ZHOU Ya-lan, Wang Jia-hai, YIN Jian
(*Department of Computer Science, Sun Yat-Sen University, Guangzhou, Guangdong* 510275, *China*)

Abstract: The philosophy behind the original particle swarm optimization (PSO) is to learn from individual's own experience and the best individual's experience in the whole swarm. Estimation of distribution algorithms (EDAs) generate new solutions from a probability model which characterizes the distribution of the current promising solutions in the search space. A novel discrete particle swarm optimization algorithm based on estimation of distribution (EDPSO) is proposed by reasonably combining the ideas of PSO and EDAs. The proposed algorithm breaks the confine of the original speed and location model, and each particle in the population have comprehensive learning ability. Therefore the proposed algorithm effectively extends the PSO to solve combinatorial optimization problems. Simulation results show that the proposed algorithm has superior performance to other discrete PSOs.

Key words: discrete particle swarm optimization; estimation of distribution; bipartite subgraph problem; combinatorial optimization problems

1 引言

粒子群优化 (Particle swarm optimization: PSO) 是一种基于群体智能的进化计算技术, 由美国的 Kennedy 和 Eberhart 于 1995 年提出, 其思想来源于对鸟群、鱼群等群集行为的模拟[1]. 与遗传算法 (Genetic algorithm: GA) 相比, PSO 具有记忆性, 而且原理更简单、参数更少, 更容易实现[1]. 自 PSO 提出后, 很多研究者对它进行了研究并提出了很多改进版本, 比如带有惯性权重的粒子群、合作粒子群等[2]. PSO 算法已经被成功应用于函数优化、神经网络训练、模糊系统控制以及其他工程领域, 但是这些算法以及它们的大部分应用都是针对连续优化问题的[2]. 如何将 PSO 算法应用于离散空间优化问题, 特别是一类非数值化优化问题, 是一个重要的研究方向[2].

近年来, 许多研究者提出了解决组合优化问题的离散版本 PSO, 其方法大致可以分为如下几种思想:

第一、与连续版本的速度 - 位移公式直接相关或对应. 文献[3]依据特定的离散优化问题, 在连续版本的速度 - 位移公式上重新定义特定的加减等运算. 但是这些精心定义的速度位移运算是依赖于特定问题的, 具有高度的技巧性, 不具有通用性. 文献[4]直接应用连续版本解决离散问题, 他们直接把连续值映射成整数或离散值, 比如对迭代产生的连续解进行舍尾取整生成整数解. Kennedy 和 Eberhart 也提出了一种离散版本的 PSO (Discrete version of PSO: DPSO)[5], 把连续变化的速度值映射成为取 0 - 1 的概率, 达到间接优化二进制变量的目的. 但是这种间接优化策略根据概率而非算法本身确

收稿日期: 2007-04-18; 修回日期: 2007-10-26
基金项目: 国家自然科学基金 (No. 60573097; 60773198; 60703111); 广东省自然科学基金 (No. 05200302; 06104916; 07300630); 教育部留学回国人员科研启动基金 (No. 2007-1108), 高等学校博十学科点专项科研基金 (No. 20050558017; 20070558052); 广州市科技计划项目 (2007Z3-D3071); 新世纪优秀人才支持计划资助 (No. NCET-06-0727)

定二进制变量,和直接把连续值映射成离散值的方法一样,都未能充分利用粒子群优化算法的性能.由于存在以上缺点,Hu 和粒子群算法的首先提出者之一 Eberhart 指出[2],直接把连续版本的 PSO 转换为离散版本也许不是唯一的选择.即要寻找其他有效的途径,使粒子群能够有效的处理离散优化问题,推动算法的发展和应用.

第二、利用量子特性,提出的量子粒子群算法.焦李成等将 PSO 与量子理论结合,提出了解决离散优化问题的量子粒子群算法(Quantum PSO:QPSO)[6].它吸收了量子计算中的复合位,叠加态等思想,采用量子比特为基本信息位进行个体编码.对量子个体的适应度评估方法是:首先通过随机观察方式,利用量子量测塌陷原理把量子态转换为二进制确定态,然后对转换得到的二进制确定态进行评估.

第三、依据遗传算法中交叉变异操作算子的思想,提出的粒子群算法.Yin 采用了遗传算法思想中的三路均匀交叉(3-way uniform crossoves)和按位方式变异运算,提出了解决组合优化问题的遗传粒子群算法(Genetic PSO:GPSO)[7].高亮等人根据遗传算法思想,也提出了广义粒子群优化模型[8],模型思想与 GPSO 相同但较复杂.

近年来,一些研究者从概率统计学的观点出发,将构造性模型引入进化算法,形成一类基于概率分析的新的进化算法,本文统一称为分布估计算法(Estimation of distribution algorithms:EDAs)[9,10].这些算法在每一次迭代时,首先从当前解空间选取若干 T 优质的解,然后依据提取出的优质解建立概率分布模型,再利用这个概率分布模型产生下一代解,如此反复,直至算法的终止条件.

标准 PSO 的基本原理是每个粒子不断学习个体自身的经验和群体最优个体的经验.分布估计算法的基本思想是在每一次迭代中,依据当前优质解信息的概率分布模型来产生新解.本文吸取这两种算法各自的优点提出一种基于分布估计的离散粒子群优化算法(Discrete estimation of distribution particle swarm optimization:EDPSO),用于求解组合优化问题.在提出的新算法中每个粒子依据两种信息来更新自己:一个是所有粒子个体历史最优解的统计信息;另一个是群体最优解的信息.提出的新算法突破了传统粒子群速度-位移搜索模型的局限,种群中的每个粒子具有更全面的学习能力,从而能够有效地解决组合优化问题.二分图问题是一个经典的组合优化问题,本文对该问题进行仿真实验来检验算法的有效性.同时,我们也选取另外三种具有代表性的离散粒子群算法,即离散版本 PSO(DPSO)、量子 PSO(QPSO)和遗传 PSO(GPSO),和提出的 EDPSO 加以对比,仿真结果表明新算法的性能优于 DPSO、QPSO 和 GPSO.

本文的组织如下:第二节和第三节分别简要地介绍 PSO 和 EDAs,第四节提出基于分布估计的离散粒了群优化算法,第五节是仿真实验,第六节总结全文.

2　粒子群优化算法(PSO)

PSO 首先初始化为一群随机粒子(随机解),然后通过迭代搜寻最优解.在每一次迭代中粒子通过跟踪两个极值来更新自己的速度和位置(速度—位移模型):一个是粒子本身所找到的最好解,即个体历史极值;另一个极值是整个种群目前所找到的最好解,称为全局最优值.

设种群中的粒子个数为 N,一个粒子所处的位置表示问题的一个候选解,粒子 i 的信息用 D 维向量表示,$i=1,2,\cdots,N$,在第 t 次迭代时粒子 i 的位置表示为 $X_i(t)=(x_{i1}(t),x_{i2}(t),\cdots,x_{id}(t),\cdots,x_{iD}(t))$,其中 $x_{id}(t)\in R$ 是粒子 i 在第 t 次迭代时第 d 维的位置.速度表示为 $V_i(t)=(v_{i1}(t),v_{i2}(t),\cdots,v_{id}(t),\cdots,v_{iD}(t))$,其中 $v_{id}(t)\in R$ 是粒子 i 在第 t 次迭代时第 d 维的速度,$PB_i(t)=(pb_{i1}(t),pb_{i2}(t),\cdots,pb_{id}(t),\cdots,pb_{iD}(t))$ 是到第 t 次迭代为止粒子 i 的个体历史极值的位置,$GB(t)=(gb_1(t),gb_2(t),\cdots,gb_d(t),\cdots,gb_D(t))$ 是到第 t 次迭代为止种群中所有粒子所经历过的最好位置.每个粒子根据个体历史极值和全局最优值来更新自己的速度和位置,其速度-位移模型用方程(1)(2)描述为[1]:

$$v_{id}(t+1)=v_{id}(t)+c_1\cdot r_1\cdot(pb_{id}(t)-x_{id}(t))+c_2\cdot r_2\cdot(gb_d(t)-x_{id}(t)) \tag{1}$$

$$x_{id}(t+1)=x_{id}(t)+v_{id}(t+1) \tag{2}$$

其中常量 c_1 和 c_2 为学习率,通常二者在 0～2 之间取相同的值,r_1 和 r_2 为两个在[0,1]区间上均匀分布的随机数.

上面介绍的是最初的标准 PSO 算法,主要适用于求解连续优化问题.Kennedy 和 Eberhart 也提出了一个离散版本的 PSO(DPSO)[5],DPSO 把标准 PSO 的方程(2)替换为:

$$x_{id}=\begin{cases}1, & \text{if } rand()\le s(v_{id})\\ 0, & \text{otherwise}\end{cases} \tag{3}$$

其中 $s(v_{id})=1/(1+\exp(-v_{id}))$ 为 Sigmoid 函数,$rand()$ 为[0,1]区间上均匀分布的随机数,可以发现 v_{id} 越大,x_{id} 取 1 的概率就越大,可用一个最大速度 V_{max} 限定算法所允许的概率范围.

3　分布估计算法(EDAs)

分布估计算法依据进化过程中优质解信息的概率分布模型产生新解,具有概率分析的理论基础[9,10].EDAs 的基本框架可描述如下:

第一步:$t=0$,随机产生初始种群 $X_{base}(t)$;

第二步:选择,从当前种群 $X_{base}(t)$ 中根据某种选择机制选择若干粒子构成父代种群 $X_{parent}(t)$;

第三步:建模,采用某种概率模型评估 $X_{parent}(t)$ 构建一个概率分布模型 $P(t)$;

第四步:抽样,依据概率分布模型 $P(t)$ 抽样产生新解 $X_{offspring}(t)$;

第五步:判断是否满足终止条件,若满足则第六步;否则 $t = t + 1$,转至第二步;

第六步:种群 $X_{base}(t+1)$ 为所求解.

EDAs 的核心部分是选择、建模和抽样. 它首先随机生成初始种群,然后根据某种机制选择拥有较好目标函数值的若干个体组成父代种群,再建立一个能反映父代种群的概率分布模型,根据概率分布模型抽样产生新种群. EDAs 的实质是抽取优质解的整体信息,然后评估它们的分布,再利用这种分布产生新的种群. 近年来,分布估计算法已成为算法研究中的一个热点.

4 基于分布估计的离散粒子群优化算法(EDPSO)

针对现有的离散粒子群算法复杂,不具有通用性,存在容易陷入局部极小值等缺点,我们把分布估计算法思想引入到粒子群算法中,提出一种基于分布估计的离散粒子群优化算法(EDPSO),它保持粒子群算法简单有效的特性,是一种新的解决组合优化问题的离散粒子群算法.

4.1 算法设计

在标准粒子群算法的速度方程(1)中,有三项:即惯性项、向个体自身历史最优学习项和向群体最优学习项,可分别解释为随机—局部搜索项、个体历史最优信息项和至今全局最优信息项[8]. 我们选择所有个体的历史最优信息,建立一个反映优质解分布的概率模型,这个概率模型标识解空间中最具潜力解的区域分布信息. 对于新的种群,随机从概率模型和至今全局最优信息项中获取解信息. 相应的随机—局部搜索项功能,通过变异或者通用的局部搜索算法实现.

EDAs 有几种不同的概率模型,其中单变量边缘分布(Univariate marginal distribution:UMD)模型是简单但被广泛应用的一种模型[10],它被应用在单变量边缘分布算法(Univariate marginal distribution algorithm:UMDA)[10],基于群体的增量学习算法(Population-based increased learning:PBIL)[9],和带 guided mutation 的进化算法[11]中. 因此,本文也采用一阶统计量的 UMD 概率模型来建模[10],评估整个解空间中优质解的分布情况.

在 EDPSO 算法中,第 t 次迭代时粒子 i 的位置 $x_{id}(t) \in \{0,1\}$. 首先统计个体历史极值的每一维出现 1 的个体数量,根据概率模型计算出一个实值概率向量,表示为 $P = (p_1, p_2, \cdots, p_d, \cdots, p_D)$,$p_d$ 表示粒子的第 d 维取值为 1 的概率. 然后,这个概率向量 P 引导粒子在 0 - 1 的解空间进行搜索. 在产生下一代种群时,以大概率 β 从概率向量抽样产生新解,以小概率 $1-\beta$ 直接复制种群全局最优解作为新解. 即个体粒子的部分维值由概率向量来决定,部分维值直接来源于群体最优个体.

EDPSO 算法产生新个体粒子的机制可由下图描述:

if $rand() < \beta$
  if $rand() < p_d$, set $x_{id}(t+1) = 1$,
    otherwise set $x_{id}(t+1) = 0$.
otherwise $x_{id}(t+1) = g_{bd}(t)$.

图 1 EDPSO 算法产生新个体粒子的机制

其中 $rand()$ 是在 $[0,1]$ 区间上的随机数. 每一维 x_{id} 的值是依据概率向量 P 产生还是依据全局最优值产生,由参数 β 控制.

概率向量 P 用如下的方程进行初始化:

$$p_d = \frac{\sum_{i=1}^{N} pb_{id}}{N} \tag{4}$$

在每一次迭代中,按照 PBIL 的概率更新规则更新概率向量 P[9]:

$$p_d = (1-\lambda)\, p_d + \lambda \frac{\sum_{i=1}^{N} pb_{id}}{N} \tag{5}$$

其中 $\lambda \in [0,1]$ 是学习率,式中右边的 p_d 代表上一代概率向量信息,$\frac{\sum_{i=1}^{N} pb_{id}}{N}$ 是在当代优质解上估计出的新的概率信息. 按照 PBIL 规则,算法根据概率向量 P 抽样产生下一代解,学习率 λ 平衡算法的全局探索和局部开采能力. 当学习率 λ 比较小时,算法具有较强的全局探索能力,例如,当学习率为 0 时,算法只有全局探索能力而没有局部开采能力. 随着学习率 λ 取值的增长,局部开采能力逐渐增强,全局探索能力随之减弱.

在 EDAs 中,当概率向量趋近于 0 或 1 时,产生的个体相似性会增加. 为了防止概率向量 P 的每一维过快地收敛到 0 或者 1,可在 EDAs 中引入变异机制. 在 PBIL 算法中的一种变异方法是直接在概率向量的每一维上加入小的随机扰动,增加解的多样性,这样能帮助算法发现更好地最终解,但是效果并不明显[9]. 所以,EDPSO 算法中采用的方法是直接在抽样产生的新解上执行按位变异来保持种群的多样性,避免算法的早熟. 即算法按图 1 产生新个体后,再对这些新粒子个体以一定的变异概率 α 实施按位变异,用方程描述如下:

$$x_{id}(t) = 1 - x_{id}(t), \quad \text{if } rand() < \alpha \tag{6}$$

4.2 算法步骤和分析

基于上述设计思想,我们给出 EDPSO 算法的具体流程和步骤,描述如下:

第一步:初始化种群,并保存所有粒子的个体历史

极值和全局最优值;

第二步:用方程(4)初始化概率向量 P;

第三步:根据图 1 所示的过程生成新个体粒子;

第四步:按方程(6)进行变异操作,生成新一代种群;

第五步:评价新种群中所有粒子的适应度;

第六步:比较适应度值更新当前的个体历史极值和全局最优值;

第七步:使用方程(5)更新概率向量 P;

第八步:判断是否满足终止条件,若满足则退出;否则转至第三步.

在算法中,由于采用的是简单的一阶统计量的 UMD 模型,更新概率模型和抽样产生新个体粒子的计算都是线性时间复杂度.算法的主要开销还是在粒子适应度评价上,这个开销主要依赖于算法所要解决的具体问题的目标函数的计算复杂度.可设最大循环次数作为算法的终止条件.

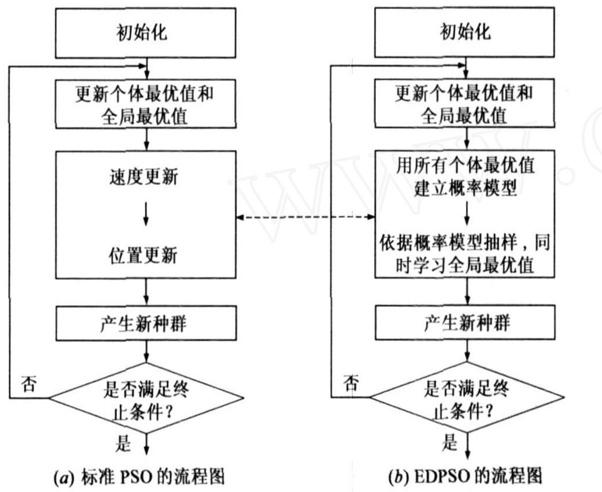

图 2 标准 PSO 的流程图和 EDPSO 的流程图

图 2(b)是新算法的流程图,与图 2(a)标准粒子群算法的流程图形成对比,从中我们可以看出新算法的特点.在新算法中,由于概率模型是选择所有个体的历史最优信息建立的,粒子的每一次更新,都向全部个体的历史最优信息学习,而不像在原始粒子群算法中粒子仅向自身个体历史最优学习,使得新算法中的粒子具有更全面的学习能力,这样能够使粒子脱出局部最小值,避免早熟.而在原始 EDAs 中,每次模型的采样都忽略了至今全局最优信息,EDPSO 算法正好弥补了这一点.因此我们提出的新算法吸取了 PSO 和 EDAs 优点的同时,又摈弃了各自的缺点.此外,在每一次迭代时,概率向量 P 依据新产生的个体粒子用 PBIL 规则来更新自己,即随着迭代的进行,高质量的个体历史最优解被选出用来更新概率向量,被更新的概率向量又用来生成新的解,这样反复进行.也就是说,始终由描述群体中的优质解(优质解包括所有粒子的个体最优值和全局最优值)的概率模型来引导整个种群的不断进化.所提新算法同时引入按位变异机制,增强种群的多样性,进一步防止算法的早熟.

5 仿真实验

为了验证提出算法的有效性,在二分图问题上进行了仿真实验,并与现有的求解组合优化问题的 QPSO、DPSO 和 GPSO 进行比较.所有的算法用 C 语言在 P4 2.80 GHz 的 PC 上实现.

5.1 二分图问题

二分图问题是一个经典的 NP-难问题[12].设 $G=(V,E)$是一个无向图,其中 V 表示顶点集合,E 表示边集合,如果图 G 的顶点集合 V 能被划分成两个不相交的集合,同一集合中的任意两个顶点之间没有边相连,那么这个图 G 被称为二分图.对于一个给定图,二分图问题的目标是从图中移走最少数量的边使剩余的图成为二分图,或者说保留最多数量的边而使剩余的图成为二分图.例如:给定一个由五个顶点和七条边组成的简单无向图如图 3(a)所示,如果移走一条边这个图就变成了如图 3(b)所示的二分图.许多实际问题都可以归结为二分图问题,如:电路布线中的最小交叉问题(Fixed linear crossing number problem:FLCNP)[13].

二分图问题的适应度函数描述如下:

$$fitness=\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}d_{ij}(x_i+x_j-2x_ix_j) \tag{7}$$

其中 d_{ij}表示顶点 i 和 j 是否有边相连,一个粒子表示为 $X_i=(x_1,x_2,\cdots,x_n)$.设二分图的两个集合分别为 S_1 和 S_2,$x_i=1$ 表示顶点 i 在集合 S_1 中,$x_i=0$ 表示顶点 i 在集合 S_2 中.

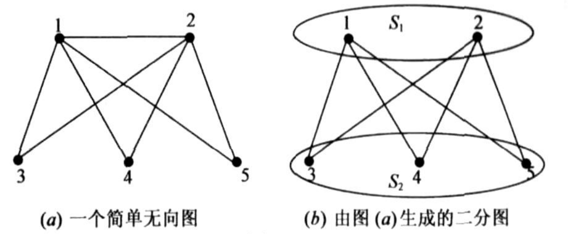

图 3

5.2 选择用于比较的算法

为了充分显示算法的性能,我们选取三种具有代表性的离散粒子群算法作对比,即在引言中提到的 DPSO、QPSO 和 GPSO,它们分别代表三种 PSO 离散化的思想.特别地,在这里我们要简要描述一下 GPSO[7]的特点.GPSO 是一个引入了遗传重组操作的新的离散版本 PSO,它采用了遗传算法中的两个遗传算子:一个是交叉算子,粒子个体的每一维是从粒子本身,个体历史极值和全局最优值得到的,即把它们作为三个父代个体进

行三路均匀交叉;另外一个是变异算子,每一维按照小概率翻转其值,也就是遗传算法中的二进制变异操作. GPSO 提出后应用在一个经典的组合优化问题 - 多边形近似问题上[7]. 实验结果表明 GPSO 能大大提高离散 PSO 求解组合优化问题的性能,而且 GPSO 的性能优于 GA 和蚁群算法(Ant colony optimization:ACO). 此外,GPSO 比 GA 和 ACO 更简单,更容易实现. 本文的目的是通过与这些典型的离散 PSO 相比较,来显示我们提出的新算法的性能.

5.3 参数设置

EDPSO 中的参数有三个:参数 β 控制产生新粒子的方式,当 β 值越大时,粒子就会越多地向所有同伴粒子学习,为了让算法具有更好地全面学习的能力,设置 $\beta = 0.9$;学习率 λ 均衡算法的全局探索能力和局部开采能力,为了让算法具有较好的搜索性能,即一方面保证算法具有一定的全局探索能力,在全局范围内探索较好的粒子,另一方面又能够使得算法在优质解周围进行精细的局部开采,设置学习率 $\lambda = 0.7$;变异率 α 是粒子发生变异的概率,变异的目的是为了防止算法陷入局部极小值,但是变异概率不能太大,否则成了随机搜索,因此变异率取较小值,$\alpha = 0.001$. 当然,参数的具体取值往往需要进行一些初步实验调整而得,而且对于不同的优化问题参数的取值可能也不同,但是因为我们已经讨论过新算法中每个参数的取值范围、意义和影响,所以既使对于不同的优化问题我们的参数取值也只需要在小范围内进行选择和调整.

用于比较的三种算法的参数按照已有文献中的方法设置,具体参数取值如下:在 QPSO[6] 中,$\alpha = 0.1$,$\beta = 0.9$,$c_1 = c_2 = 0.1$;在 DPSO[5] 中,$c_1 = c_2 = 1.2$,最大速度 $V_{\max} = 4$;在 GPSO[7] 中,w_1 的值随着迭代的进行从 0.9 降到 0.4,$w_2 = 0.2w_1 + 0.8$,变异概率 $p_m = 0.001$. 所有的算法种群规模均为 20,迭代次数为 500,分别进行 20 次独立实验.

5.4 仿真结果

随机生成 100～500 个顶点的 15 个测试图作为测试用例,具体参数如表 1 前三列所示,其中参数 p,$0 < p < 1$,表示无向图中边的密度. 表 1 的后八列是四种算法各 20 次独立实验的结果,表中的"最大值"表示每一种算法处理每一个测试图时,在 20 次独立运行中产生的最大剩余边数,表中的"平均值"表示每一种算法处理每一个测试图时,20 次独立运行产生的剩余边的平均数. 从结果可见 EDPSO 处理 15 个测试图得到的最大值和平均值比 QPSO、DPSO 和 GPSO 的都好. 甚至,EDPSO 求解每一个测试图所得的平均值都要优于 QPSO 求解对应测试图所得的最大值. 仿真结果表明新算法具有非常好的寻优性能.

表 2 是四种算法 20 次独立实验的平均运行时间,从结果可见 DPSO 的运行速度最慢,原因是它在每一次迭代时需要花费时间计算 Sigmoid 函数. 提出的 EDPSO 的运算速度稍慢于 QPSO 和 GPSO,因为该算法在每一次迭代中需要花费时间更新概率分布向量,但增加的时间开销很小. 所以,EDPSO 的总体性能要优于 QPSO、DPSO 和 GPSO.

表 1 四种算法的结果比较

顶点数 n	边数 e	p	QPSO		DPSO		GPSO		EDPSO	
			最大值	平均值	最大值	平均值	最大值	平均值	最大值	平均值
100	1485	0.3	889	870.4	906	892.6	904	891.95	908	895.75
	2970	0.6	1653	1631.7	1669	1653.9	1661	1649.15	1671	1657.05
	3465	0.8	1886	1870.3	1913	1899	1911	1894	1916	1903
200	5970	0.3	3300	3243.95	3430	3397.65	3443	3400.25	3450	3419.1
	11940	0.6	6279	6258.7	6446	6422.8	6459	6418.6	6467	6437.85
	15920	0.8	8230	8210.4	8368	8336.75	8360	8333.45	8375	8358.7
300	13455	0.3	7153	7117.35	7485	7429.95	7509	7460.35	7545	7485.45
	26910	0.6	13986	13888.4	14264	14215.9	14282	14243.1	14356	14285.2
	35880	0.8	18338	18312.8	18632	18578.3	18644	18598.7	18697	18627.3
400	23940	0.3	12559	12489.7	13011	12945.8	13121	13043.9	13137	13085.1
	47880	0.6	24564	24513.3	25123	25028.1	25197	25106	25255	25165.2
	63840	0.8	32442	32403	32880	32818.3	32947	32892.3	33030	32950.9
500	37425	0.3	19401	19355.3	20068	19999.4	20211	20160.4	20321	20242.5
	74850	0.6	38164	38128.8	38928	38826.2	39133	38999.2	39151	39062.1
	99800	0.8	50554	50506.1	51134	51051.1	51290	51217	51347	51273

表 2　四种算法的平均运行时间比较(秒)

n	e	p	QPSO	DPSO	GPSO	EDPSO
100	1485	0.3	1.56	2.73	2.37	2.41
	2970	0.6	1.60	2.77	2.26	2.40
	3465	0.8	1.68	2.74	2.26	2.37
200	5970	0.3	5.99	8.93	8.00	8.24
	11940	0.6	5.95	8.98	8.09	8.31
	15920	0.8	6.05	9.01	8.07	8.32
300	13455	0.3	13.16	18.37	17.1	17.97
	26910	0.6	13.73	18.53	17.2	17.76
	35880	0.8	14.27	18.65	17.4	17.88
400	23940	0.3	28.13	31.26	29.8	30.30
	47880	0.6	28.72	32.19	30.23	30.87
	63840	0.8	28.52	32.87	30.22	31.09
500	37425	0.3	41.05	51.29	47.03	47124
	74850	0.6	37.66	49.54	46.78	47.54
	99800	0.8	37.78	55.10	48.34	51136

单一的进化算法求解复杂组合优化问题的效果并不是特别好,这是因为单一的进化算法不具备很强的局部细搜索能力.所以本文的目的不是为了显示提出算法的性能优于其他专门求解二分图的算法或者混合算法(Memetic algorithms)[14],而是为拓展 PSO 用于组合优化问题提供一个新思路,显示 EDA 怎样来提高离散版本 PSO 的性能.而且,本文通过与其他几种求解组合优化问题的典型离散 PSO 进行对比实验,显示了 EDPSO 的相对性能.如果要进一步提高算法解决二分图问题的绝对性能,我们可以直接在算法中融合线性时间复杂度的局部搜索算法,如文献[14],形成新的 Memetic algorithms,这也是我们下一步的工作.

6　结语

本文提出了一种基于分布估计的离散粒子群优化算法(EDPSO),用于求解组合优化问题.EDPSO 突破了原始粒子群速度 - 位移模型的局限,把分布估计算法思想引入到粒子群算法中,粒子的每一次更新,都向全部个体的历史最优信息学习,而不像在原始粒子群算法中粒子仅向自身个体历史最优学习.二分图问题的仿真结果表明,EDPSO 比其他几种求解组合优化问题的典型离散粒子群算法具有更好的性能.EDPSO 为有效地把粒子群算法拓展到解决组合优化问题的领域提供了一个新的思路.

由于 EDPSO 采用 0-1 编码,所以它能很方便的用来解决其它 0-1 整数规划问题,如多边形近似问题[7].我们的进一步的工作是:(1)把 EDPSO 和局部搜索方法合理结合并应用到其他的组合优化问题中;(2)用高阶概率分布模型来评估优质解的分布信息,并尝试应用到更多的难解的优化问题中.

参考文献:

[1] Kennedy J, Eberhart R C. Particle swarm optimization[A]. Proceedings of IEEE International Conference on Neural Networks [C]. Piscataway, NJ: IEEE Press, 1995. 1942 - 1948.

[2] Hu X, Shi Y Eberhart R C. Recent advance in particle swarm [A]. Proceedings of the 2004 Congress on Evolutionary Computation[C]. Piscataway, NJ: IEEE Press, 2004. 90 - 97.

[3] 钟一文,蔡荣英.求解二次分配问题的离散粒子群优化算法[J].自动化学报,2007,33(8):871 - 874.
Zhong YW, Cai RY. Discrete particle swarm optimization algorithm for QAP[J]. Acta Automatica Sinica, 2007, 33(8):871 - 874. (in Chinese)

[4] Tasgetiren M F, Liang Y-C, Sevkli M, Gencyilmaz G. A particle swarm optimization algorithm for makespan and total flowtime minimization in the permutation flowshop sequencing problem[J]. European Journal of Operational Research, 2007, 177:1930 - 1947.

[5] Kennedy J, Eberhart R C. A discrete binary version of the particle swarm algorithm[C]. Proceedings of the World Multiconference on Systemics, Cybemetics and Informatics[C]. Bscataway. NJ, 1997. 4104 - 4109.

[6] Yang S Y, Wang M, Jiao L C. A quantum particle swarm optimization[A]. Proceeding of the 2004 IEEE Congress on Evolutionary Computation[C]. IEEE 2004, 1:320 - 324.

[7] Yin P Y. Genetic particle swarm optimization for polygonal approximation of digital curves[J]. Pattern Recognition and Image Analysis, 2006, 16(2):223 - 233.

[8] 周驰,高亮,高海兵.基于 PSO 的置换流水车间调度算法[J].电子学报,2006,34(11):2008 - 2011.
Zhou C, Gao L, Gao HB. Particle swarm optimization based algorithm for permutation flow shop scheduling[J]. Acta Electronica Sinica, 2006, 34(11):2008 - 2011. (in Chinese)

[9] Baluja S. Population-based incremental learning: A method for integrating genetic search based function optimization and competitive learning[J]. School of Comput. Sci, Carnegie Mellon Univ, Pittsburgh, PA, Tech Rep CMU-CS-94-163, 1994.

[10] Mühlenbein H. The equation for response to selection and its use for prediction[J]. Evolutionary Computation, 1997, 5(3):303 - 346.

[11] Zhang Q, Sun J, Tsang E. An evolutionary algorithm with guided mutation for the maximum clique problem[J]. IEEE Transactions on Evolutionary Computation, 2005, 9:192 - 200.

[12] Karp R M. Reducibility among combinatorial problems[M]. Complexity of Computer Computations, Plenum Press, R Miller, J Thatcher, eds, New York, 1972. 85 - 104.

[13] Buchheim C, Zheng L. Fixed linear crossing minimization by reduction to the maximum cut problem[A]. Computing and

Combinatorics, 12th Annual International Conference, COCOON 2006[C]. Lecture Notes in Computer Science, 4112: 507 - 516.

[14] Merz P, Katayama K. Memetic algorithms for the unconstrained binary quadratic programming problem[J]. BioSystems, 2004, 78: 99 - 118.

作者简介:

周雅兰 女,1979 年 3 月出生于湖南常德,中山大学博士研究生,主要研究方向为人工智能和数据挖掘.
Email: zhouylan@163.com.

王甲海 男,1977 年 6 月出生于江西石城,中山大学讲师,主要研究方向为计算智能及其应用,本文通讯作者.
Email: wangjiah@mail.sysu.edu.cn.

印 鉴 男,1968 年 4 月出生于湖北仙桃,中山大学教授,博士生导师,主要研究方向为数据挖掘和机器学习.
Email: issjyin@mail.sysu.edu.cn.