

基于混合采样机制的互信息分布估计算法

林志毅[†], 谢国波, 王丽娟

(广东工业大学 计算机学院, 广州 510006)

摘要: 针对二阶分布估计算法的早熟收敛问题, 提出一种基于混合采样机制的互信息分布估计算法 (MIEDA). MIEDA 利用互信息度量变量之间的相关性, 形成互信息树的概率模型; 采用稀疏模型构建的思想, 并基于自私基因理论建立信息奖惩机制, 以加快算法的收敛速度; 结合反向学习、最优解变异和随机采样形成混合采样机制, 以提高算法的采样效率. 仿真结果表明, MIEDA 比常见的二阶分布估计算法具有更高的稳定性和更强的寻优能力.

关键词: 互信息; 自私基因; 信息奖惩机制; 混合采样机制

中图分类号: TP18

文献标志码: A

Mutual information estimation of distribution algorithm with hybrid sampling mechanism

LIN Zhi-yi[†], XIE Guo-bo, WANG Li-juan

(Faculty of Computer, Guangdong University of Technology, Guangzhou 510006, China)

Abstract: A mutual information estimation of distribution algorithm (MIEDA) with hybrid sampling mechanism is proposed to overcome premature convergence of second order estimation of distribution algorithms. The MIEDA firstly uses mutual information to measure the interaction between two variables, which can generate mutual information tree model. Then, based on the concept of sporadic model building and a reward and punishment scheme in the selfish gene, the MIEDA can accelerate the convergence speed. Finally, a hybrid sampling mechanism is also adopted in the MIEDA to improve the efficiency of sampling, which combines stochastic sampling, the opposition-based learning (OBL) scheme and mutation on the current optimal individual. The simulation results show that, compared with several other second order algorithms, the MIEDA often performs better in convergent reliability and search ability.

Keywords: mutual information; selfish gene; reward and punishment scheme; hybrid sampling mechanism

0 引言

分布估计算法 (EDA) 是一种基于概率模型的进化算法^[1-2], 采用基因间的联合概率分布表示其连锁关系, 利用统计学习方法引导种群的进化方向, 从原理上克服了传统进化计算中难以解决的连锁学习问题. 目前, 分布估计算法已经成为进化计算领域的研究热点和解决实际工程应用问题的有效方法. 人们提出了各种不同的分布估计算法, 并在组合优化^[3-5]、多目标优化^[6-7]、机器学习^[8] 和 多分类器系统构造^[9] 等方面得到了成功的应用.

众多的分布估计算法中, 二阶分布估计算法由于模型相对简单, 求解效果尚可, 得到了一些研究者的青睐. 目前已提出了 MIMIC^[1]、COMIT^[1]、BMDA^[1]、

SGMIEC^[10] 等二阶分布估计算法, 并在函数优化和组合优化方面得到了应用. 然而, 目前的二阶分布估计算法存在以下两个问题:

1) 在建立概率模型时, 需要统计大量的群体样本才能得到较为准确的概率模型信息, 导致群体规模对模型的影响很大, 如果种群规模较小, 则建立的概率模型可能无法真正描述群体的信息, 从而导致算法的求解效果不好. 此外, 如果每一代都建立概率模型, 算法的时空开销则会较大.

2) 在对概率模型采样时, 基本上都是采用单一的采样机制且大都为随机采样. 然而, 到目前为止, 没有机制保持采样时的种群多样性, 也没能充分利用当前已发现的最优个体的局部信息.

收稿日期: 2016-04-16; 修回日期: 2016-07-06.

基金项目: 国家自然科学基金项目 (61502108); 广东省重大科技专项项目 (2014B010111007); 广东省自然科学基金项目 (2014A030313512); 广东省公益研究与能力建设专项项目 (2016A010101027).

作者简介: 林志毅 (1979-), 男, 讲师, 博士, 从事自然计算、信息安全的研究; 谢国波 (1977-), 男, 教授, 博士, 从事信息安全、云计算等研究.

[†]通讯作者. E-mail: lzy291@gdut.edu.cn

针对上述问题,本文针对二阶分布估计算法展开研究,提出一种新的基于混合采样机制的互信息分布估计算法(MIEDA),并通过实验验证MIEDA的有效性和稳定性.

1 基于混合采样机制的互信息分布估计算法

MIEDA的主要思想启发来源于COMIT和自私基因理论.下面分别就MIEDA中概率模型的构建、概率模型的更新和混合采样机制进行阐述,然后给出算法流程.

1.1 概率模型的构建

MIEDA使用一个频率矩阵统计个体信息的频率和出现的位置,然后根据该频率矩阵计算各变量之间的互信息和自信息,进而构建概率关系模型.

1.1.1 关系度量标准

MIEDA采用互信息作为变量关系度量的标准,两个变量的互信息 $I(X;Y)$ 表示为

$$I(X;Y) = \sum_{x \in X} \sum_{y \in Y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)}, \quad (1)$$

满足 $I(X;Y) \geq 0$.当 X 和 Y 为同一个变量时,式(1)为自信息.

1.1.2 基于互信息的概率模型构建

MIEDA采用与COMIT相同的概率模型,如图1所示.

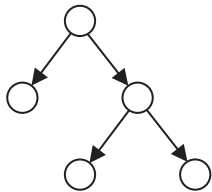


图1 MIEDA树状结构的概率模型

为了进行概率模型的构建,MIEDA设置一个频率矩阵 F ,其中 $F[x_i = a, x_j = b]$ 用于记录变量对 x_i 和 x_j 中出现 $x_i = a$ 且 $x_j = b$ 的次数.首先将 $F[x_i = a, x_j = b]$ 初始化为一个小常量,然后根据优势群体中的信息更新数组 F ,即每出现一个 $x_i = a$ 且 $x_j = b$, $F[x_i = a, x_j = b]$ 的值增加1.在 F 构造完毕后,可根据式(1)计算得到每个变量的自信息和两个变量对的互信息.假设全体变量集合为 $K = \{x_1, x_2, \dots, x_n\}$,变量个数为 n ,则生成概率模型的算法如下所示.

算法1 概率模型生成算法.

Step 1: 计算每个变量 $x_k (1 \leq k \leq n)$ 的自信息,选择自信息最小的变量 x_r 作为树的根结点.设置一个parent数组以指示每个变量的父变量,并令 $\text{parent}(x_i) = x_r (1 \leq k \leq n \text{ 且 } k \neq r), \text{parent}(x_r) =$

$0, S = \{x_r\}, V = K - \{x_r\}$.

Step 2: 从 V 与 S 之间选择互信息最大的变量对 (x_i, x_j) ,其中 $x_i \in V, x_j \in S$.将 x_j 加入到集合 S 中,并将其从 V 中移出,即 $S = \{x_j\} \cup S, V = V - \{x_j\}$,令 $\text{parent}(x_i) = x_j$.

Step 3: 若 $V \neq \emptyset$,则转至Step 2,否则结束.

1.2 概率模型的更新

为了减少概率模型构建的时空开销,MIEDA降低了概率关系模型的构建频率,在第一次概率关系模型构建完毕后,概率关系模型采用稀疏构建模型的方式进行构建,每隔一定的代数 m 才对概率模型进行重新构建^[11],其他时间内则在采样后利用信息奖惩机制对频率矩阵 F 进行调整,以减少种群规模对概率模型的影响,从而在一定程度上解决二阶分布估计算法存在的第1个问题.

自私基因理论是Dawkins^[12]提出的一种演化观,即自然选择的基本单位是基因,不是物种,也不是种群或群体.该理论表明,个体只是作为基因的载体,个体之间的竞争其实是基因的竞争,个体的选择实际上是基因的选择,而种群的作用是作为基因池,包含了所有已知的遗传信息.因此,从自私基因理论的角度而言,传统的分布估计算法只利用了优势个体的基因,忽略了劣势个体所包含的基因.那么在进化过程中,为了同时利用优劣个体的基因,MIEDA利用种群中的最优个体 g_{best} 和最差个体 g_{worst} 的信息对频率矩阵 F 的相应基因位进行更新,从而构成了信息奖惩机制,其实现如算法2所示.

算法2 信息奖惩机制.

reward_penalize_genes($g_{\text{best}}, g_{\text{worst}}$)

for $i = 1$ to n do

for $i = 1$ to n do

$$F(g_{\text{best}}[i], g_{\text{best}}[j]) = F(g_{\text{best}}[i], g_{\text{best}}[j]) + \varepsilon;$$

$$F(g_{\text{worst}}[i], g_{\text{worst}}[j]) = F(g_{\text{worst}}[i], g_{\text{worst}}[j]) - \varepsilon;$$

end

end

奖惩因子 ε 可设置为

$$\varepsilon = \frac{2 \times \text{PopSize}}{n \times (n - 1)}. \quad (2)$$

其中:PopSize为种群的大小,在本文中其值为随机采样的个体数 N ; n 为变量的个数.

信息奖惩机制通过对频率矩阵的调整实现对概率模型的修正,一定程度上避免了采样规模较少所导致的概率模型不准确的问题,并使概率模型不会快速趋同,从而提高算法的求解能力.

1.3 混合采样机制

为了解决二阶分布估计算法存在的第2个问题,避免早熟收敛现象的出现,本文提出一种混合采样机制. 该机制结合随机采样、反向学习和最优解变异等多种采样方法以采样生成新种群,充分利用了全局统计信息和最优个体的局部信息,从而避免采样后个体差异过小的问题,使MIEDA保持较好的种群多样性. 混合采样机制的流程如图2所示.

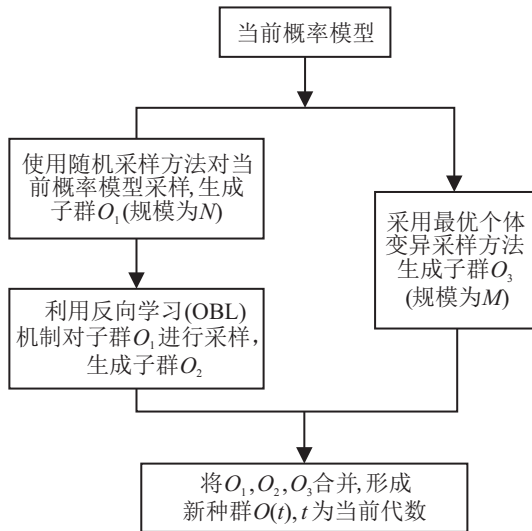


图2 混合采样机制流程

1.3.1 随机采样

随机采样按照如下算法3的思想,对变量依次采样,构造出一个完整的解向量.

算法3 随机采样生成一个新个体的算法.

Step 1: 设置 $K = V = \{x_1, x_2, \dots, x_n\}$.

Step 2: 根据根结点 x_r 的概率分布 $p(x_r)$ 随机采样生成 x_r 的值, 设置 $K = K - \{x_r\}$.

Step 3: 如果 $K \neq \emptyset$, 则转至 Step 4, 否则算法结束.

Step 4: 若 $\text{parent}(x_i) = x_j$ 且 x_j 在集合 $V - K$ 中, 则从 K 中选择 x_i .

Step 5: 根据条件概率 $p(x_i|x_j)$ 随机采样生成 x_i 的值.

Step 6: 将 x_i 从集合 K 删除, 即 $K = K - \{x_i\}$, 转至 Step 3.

反复执行 N 次算法3, 产生 N 个新个体.

1.3.2 反向学习采样

反向学习 (OBL) 是 Tizhoosh^[13] 提出的一种优化策略, 广泛应用于许多优化计算领域. 事实上, 同时评估一个候选个体 $X = \{x_1, x_2, \dots, x_n\}$ 和其对应的反向个体 $\bar{X} = (\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n)$ 更有可能找到接近全局最优解的候选个体^[14], 其中 \bar{x}_i 定义如下:

$$\bar{x}_i = \begin{cases} 1 - x_i, & x_i \in \{0, 1\}; \\ a_i + b_i - x_i, & x_i \in R, x_i \in [a_i, b_i]. \end{cases} \quad (3)$$

MIEDA 在对概率模型随机采样后引入 OBL 机制, 为每个随机采样得到的个体产生对应的反向个体, 进而从采样个体及其反向个体中挑选优秀个体进行概率模型的构建. 通过这种方式可以覆盖更多的搜索空间, 在一定程度上增加种群的多样性, 并改善优化过程的收敛速率.

1.3.3 最优个体变异采样

与种群中其他个体相比, 当前最优个体与全局最优解的距离最近, 那么通过变异操作来利用当前最优个体的信息是一种很自然的想法. 变异可以改变旧的基因, 引入新的基因到基因池中, 从而增加种群的多样性. 最优个体变异采样方法以概率 P_m 进行如下变异操作:

$$\text{mutation}(x_i) = \begin{cases} 1 - x_i, & x_i \in \{0, 1\}; \\ \text{rand}(a_i, b_i), & x_i \in R, x_i \in [a_i, b_i]. \end{cases} \quad (4)$$

其中: x_i 为随机选择的一个基因位, $\text{rand}(a_i, b_i)$ 为 $[a_i, b_i]$ 内一个随机实数. 采用上述最优个体变异的采样方法后, MIEDA 更有可能在保持当前最优个体重要信息的同时获得更优秀的个体.

事实上, 在算法搜索的前期, 当前最优个体离全局最优解还较远, 因此, 最优个体变异的采样个体数 M 应设置为一个较小的数, 避免当前最优个体信息的迅速传播, 导致算法早熟收敛. 变异概率 P_m 则应设置为一个较小的数, 在对种群多样性影响较小的情况下, 可以一定程度上加快算法的收敛速度. 此外, 在算法搜索后期, 当前最优个体离全局最优解较近, 这时种群多样性也较低, 因此, 采样个体数 M 应设置为一个较大的数, 变异概率 P_m 也需同时增加, 从而增加种群多样性, 加快收敛速度. 基于上述讨论, 本文均采用动态自适应调整的方式获得 M 和 P_m 值, 即

$$M = \frac{t}{5 \times t_{\max}} \times N, \quad P_m = 0.5 \times \frac{t}{t_{\max}}. \quad (5)$$

其中: t 为当前的进化代数, t_{\max} 为最大的进化代数, N 为随机采样的个体数. 由式(5)和(6)可见, M 和 P_m 将在整个搜索过程中动态自适应调整.

综合上述过程, MIEDA 的整体流程见算法4.

算法4 MIEDA 算法流程.

Step 1: 初始化频率矩阵 F 、种群 $P(0)$ 、随机采样个体数 N 和间隔频率 m 等参数, 并设置当前代数

$t = 0$.

Step 2: 从 $P(t)$ 中选择优势群体 $S(t)$.

Step 3: 根据 $S(t)$ 更新频率矩阵 F .

Step 4: 判断 $t \% m$ 是否等于 0, 如果是, 则根据算法 1 构造概率模型 G , 否则使用算法 2 调整频率矩阵 F .

Step 5: 使用混合采样机制从概率模型 G 中采样得到新的种群 $O(t)$, 并评估 $O(t)$.

Step 6: 使用 $O(t)$ 的前 N 个个体替换 $P(t)$.

Step 7: 判断结束条件, 如果不满足则转至 Step 4, 否则执行 Step 8.

Step 8: 算法结束, 返回最好个体的值, 或者根据情况返回所有个体的值.

2 实验分析

2.1 实验设置与测试函数

为了说明算法的性能, 本节求解 6 个常用于评价分布估计算法性能的测试函数. DEC 函数的规模为 $9 \sim 207$, 步长为 18, 其余函数规模为 $10 \sim 200$, 步长为 10. 同时将本文算法 MIEDA 与其他几个双变量相关的分布估计算法 (包括 BMDA、COMIT、MIMIC 和 SGMIEC) 进行比较. COMIT、MIMIC 和 BMDA 算法参数设置如下: 算法样本 (即种群) 大小设置为 200, 样本选择采用截断选择, 选择比为 50%. SGMIEC 参数设置如下: 样本大小为 7, 只选择最好和最差的个体作为操作对象, 每次迭代的学习参数 $\alpha = 0.1$. MIEDA 参数设置如下: 随机采样个体数 $N = 20$, 最大进化代数 $t_{\max} = 10\,000$, 间隔频率 $m = 5$. 所有实验中, 参数保持不变, 对于每个函数和函数规模, 算法都独立运行 20 次, 每次迭代终止的条件为找到全局最优或者评价次数达到操作上限 (200 000). 测试函数如下:

1) OneMax 函数. 有

$$\max \left(\sum_{i=1}^n x_i \right), x_i \in \{0, 1\}. \quad (6)$$

OneMax 函数的特点是变量无关.

2) Quadratic 函数. 有

$$\max \left(\sum_{i=1}^n f(x_{2i-1}, x_{2i}) \right), x_{2i-1}, x_{2i} \in \{0, 1\}, \quad (7)$$

其中

$$f(u, v) = 0.9 - 0.9(u + v) + 1.9uv. \quad (8)$$

Quadratic 函数的特点是双变量相关且没有重叠.

3) Four Peaks 函数. 给定 D 维输入向量 \mathbf{X} , Four Peaks 函数定义如下:

$$f(\mathbf{X}, T) = \max[\text{tail}(0, \mathbf{X}), \text{head}(1, \mathbf{X})] + R(\mathbf{X}, T). \quad (9)$$

其中

$$\text{tail}(b, \mathbf{X}) = \text{number of trailing } b's \text{ in } \mathbf{X}; \quad (10)$$

$$\text{head}(b, \mathbf{X}) = \text{number of leading } b's \text{ in } \mathbf{X}; \quad (11)$$

$$R(\mathbf{X}, T) = \begin{cases} D, & \text{tail}(0, \mathbf{X}) > T, \wedge \text{head}(1, \mathbf{X}) > T; \\ 0, & \text{otherwise.} \end{cases} \quad (12)$$

实验中, T 设置为 $10\%D$.

4) Six Peaks 函数. Six Peaks 函数与 Four Peaks 函数定义相同, 除了

$$R(\mathbf{X}, T) = \begin{cases} D, & \text{tail}(0, \mathbf{X}) > T, \wedge \text{head}(1, \mathbf{X}) > T \text{ or} \\ & \text{tail}(1, \mathbf{X}) > T, \wedge \text{head}(0, \mathbf{X}) > T; \\ 0, & \text{otherwise.} \end{cases} \quad (13)$$

Four Peaks 和 Six Peaks 函数的特点是变量之间的关系是链状的, 链的两端是相互重叠的.

5) Deceptive function of order 3 (DEC-3) 函数. 有

$$\max \left(\sum_{i=1}^{n/3} f(x_i, x_{i+1}, x_{i+2}) \right), x_i \in \{0, 1\}. \quad (14)$$

其中

$$f(u, v, w) = \begin{cases} 1, & u + v + w = 3; \\ 0, & u + v + w = 2; \\ 0.8, & u + v + w = 1; \\ 0.9, & u + v + w = 0. \end{cases} \quad (15)$$

DEC 函数的特点是 x_i 与另外两个变量相关, 并存在模式欺骗性.

6) Trap function of order 5 (Trap-5) 函数. 有

$$\max \left(\sum_{i=1}^{n/5} f(x_{5i-4} + x_{5i-3} + x_{5i-2} + x_{5i-1} + x_{5i}) \right). \quad (16)$$

其中

$$x_i \in \{0, 1\}, \quad f(u) = \begin{cases} 5 - u, & u < 5; \\ 5, & \text{otherwise.} \end{cases} \quad (17)$$

Trap-5 函数的特点是多变量相关.

2.2 实验结果及分析

首先, 测试并对比 5 个算法的收敛能力, 方法是在不同函数规模条件下, 比较最终的收敛结果与已知最优解之差, 结果见图 3. 由图 3 可见, 在所有的数值实验中, 对于 6 个测试函数, BMDA、MIMIC 和 COMIT

算法的结果都不能令人满意,在大多数情况下,3个算法求得平均最优解与实际最优解的差值要比 MIEDA 算法和 SGMIEC 算法的差值大. 比如, BMDA 算法和 COMIT 算法对于 Six Peaks 函数,在函数规模为 200 时求得平均最优解与实际最优解差值已经超过 300; MIMIC 算法对于简单的 OneMax 函数,在规

模为 200 时所求得平均最优解与实际最优解差值也达到 50. 由图 3 同样可见,在大部分函数上, MIEDA 算法与 SGMIEC 算法收敛能力相当,但在 Four Peaks 和 Six Peaks 函数上,函数规模超过 100 维后, MIEDA 算法的收敛能力明显要高于 SGMIEC 算法. 从整体收敛能力而言, MIEDA 算法要优于其他 4 个算法.

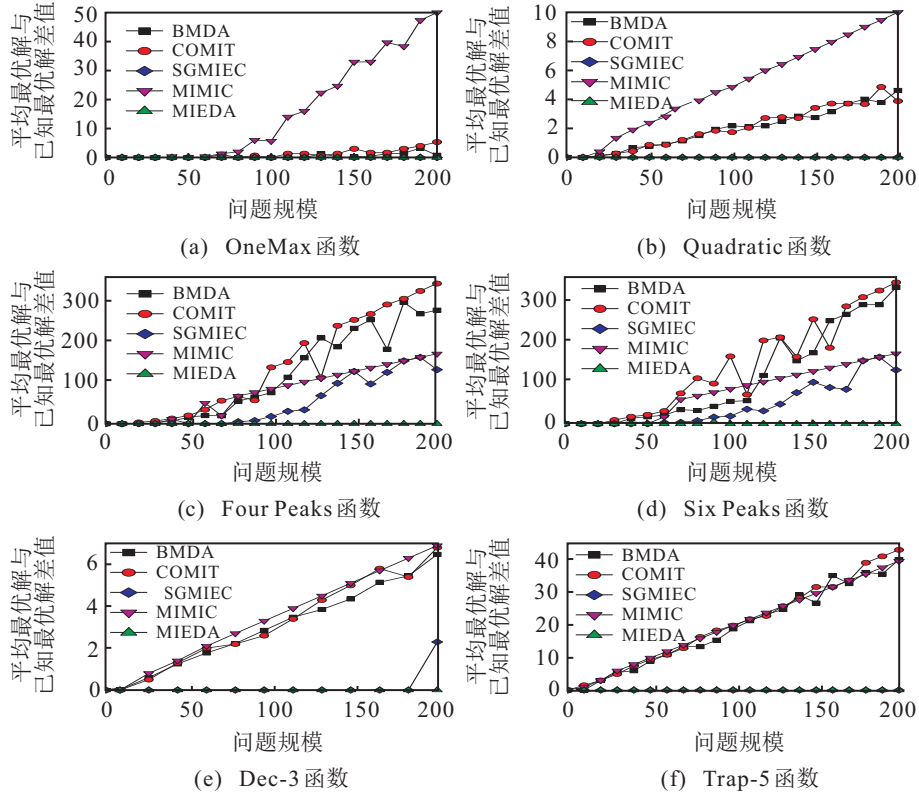


图 3 算法收敛能力比较

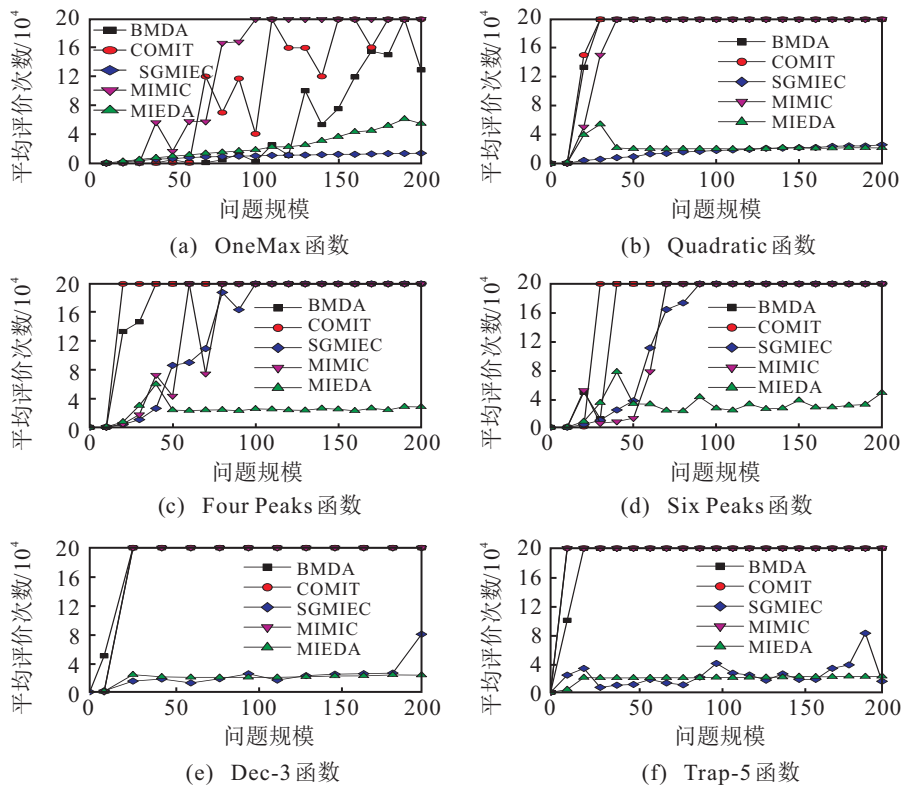


图 4 算法收敛速度比较

然后,对比5个算法的收敛速度,即达到收敛所需要的平均评价次数,实验结果见图4.由图4可见,MIEDA算法只需要较少的评价次数即可达到甚至超过BMDA、MIMIC和COMIT算法的性能.特别是当函数的规模增大时,这种差距显得更为明显.比如,在解决Trap函数时,BMDA、MIMIC和COMIT这3个算法表现都相当不理想,在函数维度仅为20时都没有收敛到全局最优解,且所需要的评价次数随着规模的增大变得非常庞大.虽然SGMIEC算法在所测试的6个函数上,所需平均评价次数与MIEDA算法相当,但其在Four Peaks和Six Peaks函数上,所需的平均评价次数较多,特别是函数规模较大时,评价次数增加很快.由图4同样可见,MIEDA算法受到规模的影响要比SGMIEC算法小,虽然在低维时,其表现比SGMIEC算法略差,但在维数高于50的情况下,所需的平均评价次数基本上没有很大的变化.值得一提的是,MIEDA算法与COMIT算法采用相同的概率模型,且COMIT算法也使用了随机采样,但MIEDA算法的平均评价次数要比COMIT算法小得多,这意味着MIEDA算法的概率模型更新机制和混合采样机制能够有效地提高算法的采样效率,极大地减少了算法的采样次数.

综合图3和图4可见,BMDA、MIMIC和COMIT算法在收敛能力和收敛速度上的曲线波动性均比较

大,意味着算法不稳定;SGMIEC算法对大部分测试函数表现较为稳定,但在Four Peaks和Six Peaks函数上表现出较大的波动,在维数不高时,所需评价次数出现明显波动,而在维数稍高的情况下则难以得到全局最优解,表明其稳定性较差;对于所有测试函数,MIEDA算法的平均最优解与已知最优解的差值都非常接近或等于0,且算法所需的平均评价次数并未随着问题规模的增加而出现明显的波动,表明其稳定性非常好.

最后,对比5个算法的收敛过程,结果如图5所示. BMDA、COMIT和MIMIC算法在求解这些函数时常常因为陷入局部最优解而失败,这可能是由于随着算法的执行,决定种群样本的概率模型开始偏向一个确定的值,导致采样后的种群越来越相似,这种相似又进一步导致概率模型趋于稳定,如此循环下去,算法便陷入了局部最优. 尽管MIEDA算法的收敛过程不是最快,特别是在算法的前期,但是它利用新的概率模型更新机制和混合采样机制使得概率模型不会快速趋同,成功地保持了种群的多样性,因此在算法的中后期表现出持续的优化能力,避免了早熟收敛问题. 此外,MIMIC算法是其中表现最差的一个算法,几乎对所有的函数,其性能都要比其他算法差,而SGMIEC算法与MIEDA算法相当.

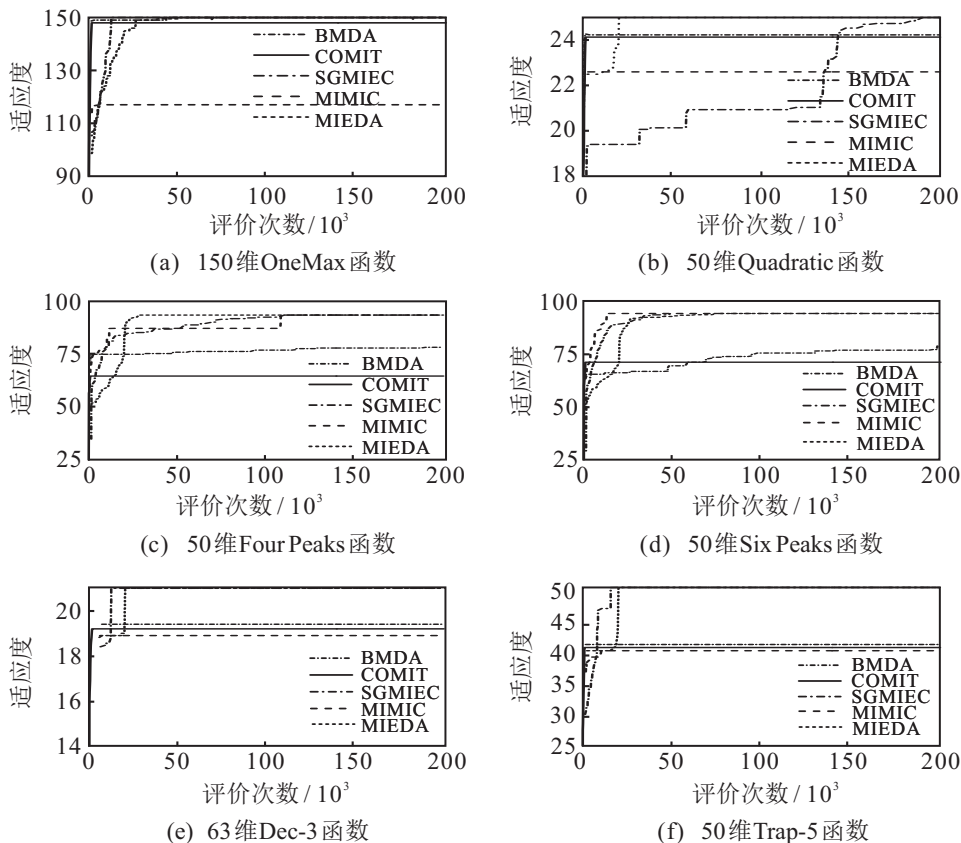


图5 算法收敛过程比较

综合表明, MIEDA 能较好地求解上述函数, 表现出了比其他4个算法更好的求解能力和稳定性, 表明其所做的改进是有效的。

3 结 论

本文提出了一种新的分布估计算法——MIEDA 算法。该算法首先利用信息论中的互信息作为变量之间的关系度量, 以频率矩阵作为信息的统计方式, 并将它们用于概率模型的生成; 然后, 以“间隔构建”的方式更新概率模型, 并从自私基因理论出发设计信息奖惩机制, 能有效地调整频率矩阵, 从而实现概率模型的修正, 减少算法的开销; 最后, 采用多种采样方式结合的方法, 充分利用种群已有的信息, 提高了算法跳出局部最优解的能力。MIEDA 算法可以较好地保持“种群多样性”和“选择压力”之间的平衡, 提高了算法的整体性能。仿真结果表明, MIEDA 算法在常见的测试函数方面取得了较好的效果, 与其他4个二阶分布估计算法相比, 具有更好的稳定性和收敛速度, 是一种比较高效的二阶分布估计算法。下一步的研究目标是算法的参数设置以及算法在其他优化领域的应用; 此外, 引入信息论或其他领域的知识从而进一步完善新算法也将是重点。

参考文献(References)

- [1] Pelikan M, Hauschild M W, Lobo F G. Estimation of distribution algorithms[M]. Springer Handbook of Computational Intelligence. Berlin: Springer-Heidelberg, 2015: 1140-1148.
- [2] 王圣尧, 王凌, 方晨, 等. 分布估计算法研究进展[J]. 控制与决策, 2012, 27(7): 961-966.
(Wang S Y, Wang L, Fang C, et al. Advances in estimation of distribution algorithms[J]. Control and Decision, 2012, 27(7): 961-966.)
- [3] Ceberio J, Irurozki E, Mendiburu A, et al. A review on estimation of distribution algorithms in permutation-based combinatorial optimization problems[J]. Progress in Artificial Intelligence, 2012, 1(1): 103-117.
- [4] Ceberio J, Irurozki E, Mendiburu A, et al. A distance-based ranking model estimation of distribution algorithm for the flowshop scheduling problem[J]. IEEE Trans on Evolutionary Computation, 2014, 18(2): 286-300.
- [5] 肖世昌, 孙树栋, 杨宏安. 混合分布估计算法求解随机Job shop 提前/拖期调度问题[J]. 控制与决策, 2015, 30(10): 1854-1860.
(Xiao S C, Sun S D, Yang H A. Hybrid estimation of distribution algorithm for stochastic Job shop scheduling with earliness/tardiness penalty[J]. Control and Decision, 2015, 30(10): 1854-1860.)
- [6] Wang L, Fang C, Suganthan P N, et al. Solving system-level synthesis problem by a multi-objective estimation of distribution algorithm[J]. Expert Systems with Applications, 2014, 41(5): 2496-2513.
- [7] Zhang J Q, Xu F, Fang X W. Decomposition of multi-objective evolutionary algorithm based on estimation of distribution[J]. Applied Mathematics & Information Sciences, 2014, 8(1): 249-254.
- [8] 曹鹏, 李博, 栗伟, 等. 基于概率分布估计的混合采样算法[J]. 控制与决策, 2014, 29(5): 815-820.
(Cao P, Li B, Li W, et al. Hybrid sampling algorithm based on probability distribution estimation[J]. Control and Decision, 2014, 29(5): 815-820.)
- [9] Mendialdua I, Arruti A, Jauregi E, et al. Classifier subset selection to construct multi-classifiers by means of estimation of distribution algorithms[J]. Neurocomputing, 2015, 157: 46-60.
- [10] Wang Feng, Lin Zhiyi, Yang Cheng, et al. Using selfish gene theory to construct mutual information and entropy based cluster for optimization[J]. Soft Computing, 2011, 15(5): 907-915.
- [11] Pelikan M, Sastry K, Goldberg D E. Sporadic model building for efficiency enhancement of the hierarchical BOA[J]. Genetic Programming and Evolvable Machines, 2008, 9(1): 53-84.
- [12] Dawkins R. The selfish gene[J]. J of Evolutionary Biology, 1989, 24(8): 1801-1813.
- [13] Tizhoosh H R. Opposition-based learning: A new scheme for machine intelligence[C]. Int Conf on Computational Intelligence for Modeling Control and Automation. Vienna: IEEE Xplore, 2005: 695-701.
- [14] Shahryar Rahnamayan, Gary Wang G, Mario Ventresca. An intuitive distance-based explanation of opposition-based sampling[J]. Applied Soft Computing, 2012, 12(9): 2828-2839.

(责任编辑: 郑晓蕾)