

二阶卡尔曼滤波分布估计算法

钟伟才 刘 静 刘 芳 焦李成

(西安电子科技大学雷达信号处理国家重点实验室 西安 710071)

摘 要 分布估计算法由于其较强的理论基础已成为进化计算研究的新热点. 从卡尔曼滤波的角度来看, 它的作用实际上是一个递归滤波器, 但作用在一个种群上的分布估计算法相当于只有一个信息源. 因此, 该文利用信息融合的思想, 将种群分成若干子种群, 各子种群独立地使用二阶分布估计算法来估计其状态, 这样就可从多个信息源获得信息. 然后用卡尔曼滤波器将这多个信息源的信息相融合, 以产生更准确的估计, 并将估计信息反馈到各子种群中. 实验结果表明, 相对于已有的二阶分布估计算法, 该文算法的稳定性和全局搜索能力都得到了很大提高, 从而说明了该文算法的有效性.

关键词 进化计算; 卡尔曼滤波; 分布估计算法; 信息融合

中图法分类号 TP301

Second Order Estimation of Distribution Algorithms Based on Kalman Filter

ZHONG Wei-Cai LIU Jing LIU Fang JIAO Li-Cheng

(Key Laboratory for Radar Signal Processing, Xidian University, Xi'an 710071)

Abstract Estimation of Distribution Algorithms (EDAs) are new evolutionary algorithms based on probabilistic model and have become a new focus in the field of evolutionary computation. From the view point of Kalman filter, EDA actually is a filter with single sensor, so its stability is poor and it is prone to be trapped in the local optima of the objective functions. To overcome these disadvantages, authors enhance its performance with Kalman filtering technique and propose a new algorithm, second order estimation of distribution algorithm based on Kalman filter. In this method, population is divided into several sub-populations, and a second order EDA for each sub-population is used to estimate the information of its state. Then, a Kalman filter is used to fuse the information so that more accurate state can be obtained. Finally, the information fused is fed back to each sub-population. Experimental results demonstrate that the algorithm outperforms available second order algorithm greatly both in the stability and the global search ability.

Keywords evolutionary computation; Kalman filter; estimation of distribution algorithm; information fusion

1 引 言

分布估计算法 (Estimation of Distribution Al-

gorithms, EDAs) 是一类基于概率模型的进化算法, 与传统进化算法不同, 它不使用交叉和变异算子, 而是根据当前种群的概率分布来产生下一代, 即先从前种群 $Pop(k)$ 中按照一定的选择方式选出一部

收稿日期: 2003-04-07; 修改稿收到日期: 2003-05-10. 本课题得到国家自然科学基金重点项目 (60133010) 资助. 钟伟才, 男, 1977 年生, 博士研究生, 研究方向为进化计算、数据挖掘、多媒体应用、模式识别. E-mail: neouma@163.com. 刘 静, 女, 1977 年生, 博士研究生, 研究方向为进化计算、数据挖掘、视频压缩、机器学习. 刘 芳, 女, 1963 年生, 硕士, 教授, 研究方向为人工智能、模式识别、进化计算. 焦李成, 男, 1959 年生, 博士, 教授, 博士生导师, 研究方向为进化计算、神经网络、子波理论.

分较好的个体 $\text{Pop}^s(k)$, 并建立其概率分布模型, 然后利用所建的分布模型产生新的个体, 以形成下一代种群 $\text{Pop}(k+1)$, 如此迭代, 直到满足终止条件. 由于 EDAs 较传统进化算法有更强的理论基础, 已成为当前进化计算的研究热点^[1~12].

设 $\mathbf{x}=(x_1, x_2, \dots, x_n)$ 为问题的解向量, $p^s(\mathbf{x}, k)$ 为当前种群的实际分布, 文献[13]已证明如果下一代种群分布 $p(\mathbf{x}, k+1)=p^s(\mathbf{x}, k)$ 且选择方式为比例、截断或锦标赛选择时, 则 EDAs 收敛于全局最优解. 因此, EDAs 的关键问题就是使 $p(\mathbf{x}, k+1)$ 在合理的计算代价下尽可能地逼近 $p^s(\mathbf{x}, k)$. 为此人们提出了各种算法, 按照其处理变量间相互作用的能力, 可分为一阶分布估计算法, 如 PBIL^[1] (Population-Based Incremental Learning), UMDA^[2] (Univariate Marginal Distribution Algorithm) 等; 二阶分布估计算法, 如 MIMIC^[3,4] (Mutual Information Maximization for Input Clustering), COMIT^[5] (Combining Optimizers with Mutual Information Trees), PADA^[6] (Polytree Approximation of Distribution Algorithm), BMDA^[7] (Bivariate Marginal Distribution Algorithm) 等; 高阶分布估计算法, 如 FDA^[8] (Factorized Distribution Algorithm), BOA^[9] (Bayesian Optimization Algorithm), BEA^[10] (Bayesian Evolutionary Algorithm), IDEA^[11] (Iterated Density Estimation Evolutionary Algorithm) 等. 一阶分布估计算法简单、快速, 但没有考虑变量间的相关性, 所以容易陷入局部极值. 显然, 高阶分布估计算法处理变量间相关性的能力最强, 但其概率分布模型非常复杂, 计算量很大, 限制了它的用途. 二阶分布估计算法的计算量比一阶略大, 同时又具有一定的描述变量间相关性的能力, 因此具有更大的实用价值.

本文从全新的角度——卡尔曼滤波来看待分布估计算法, 如果把优化问题看作一个系统, 最优解向量则可看作系统的真实状态, 优化问题的求解过程就变成了对系统进行观测以获得其真实状态的过程. 因此 EDAs 实际上就是一个递归滤波器, 其输入序列为第 k 代的种群 $\text{Pop}(k)$ 和系统状态 $\bar{\mathbf{x}}(k)$, 输出序列为预测的系统状态 $\bar{\mathbf{x}}(k+1)$, 它的动力学方程可以写为

$$\bar{\mathbf{x}}(k+1) = \text{EDAs}(\text{Pop}(k), \bar{\mathbf{x}}(k)) \quad (1)$$

由于对系统的观测存在误差, 测量到的数据(种群 $\text{Pop}(k)$)总是存在噪声, 因此估计出来的 $\bar{\mathbf{x}}(k+1)$ 也是不精确的. EDAs 的作用就在于对系统建立合适的模型并不断对其进行观测, 从而逐渐消除噪声, 获得

系统的真实状态, 即问题的最优解. 输入聚类互信息最大化算法 (Mutual Information Maximization for Input Clustering, MIMIC) 是一个典型的二阶分布估计算法, 首先由 De Bonet^[3] 等人提出并由 Larranage^[4] 等人把它推广到连续域上. 因此, 本文将 MIMIC 与卡尔曼滤波相结合, 提出了二阶卡尔曼滤波分布估计算法 (Mutual Information Maximization for Input Clustering based on Kalman Filter, KFMIMIC), 它既具有简单快速的优点, 又有较强的处理变量间相互作用的能力. 实验结果表明本文算法性能稳定, 具有很强的全局寻优能力.

2 二阶卡尔曼滤波分布估计算法

分布估计算法的作用实际上相当于一个递归滤波器, 它对观测到的数据进行滤波以获得系统的真实状态, 但作用在一个种群上的分布估计算法相当于只有一个信息源. 从信息融合的角度来看, 融合不同信息源的信息, 可产生比单个信息源更可靠、更准确的信息. 因此, 我们将种群分成若干子种群, 各个子种群独立的使用分布估计算法来估计其状态, 这样就可从多个信息源获得信息. 然后, 用卡尔曼滤波器将这多个信息源的信息相融合, 以产生更准确的估计, 并将估计信息反馈到子种群中.

2.1 卡尔曼滤波

卡尔曼滤波是一种利用所观测的时间序列来估计未知变量的方法, 它的哲学思想来自于虽然真实世界是不可知的, 但可以通过观测来逼近真实世界. 对于感兴趣的变量, 可利用多个传感器进行测量, 而卡尔曼滤波则可把这多个信息源的数据“合并”起来以得到这些感兴趣变量的最佳估计.

以一个简单的例子说明卡尔曼滤波的工作原理^[13]. 假设一艘船迷失了方向停在海上, 需要利用星星的位置来估计这艘船当前的位置. 在 t_1 时刻, 测得结果为 z_1 , 误差为 σ_1^2 , 根据条件概率密度, 船当前位置的最佳估计和估计误差的方差分别为

$$\bar{\mathbf{x}}(t_1) = z_1 \quad (2)$$

$$\sigma_{\bar{\mathbf{x}}}^2(t_1) = \sigma_1^2 \quad (3)$$

假设在 $t_2 \cong t_1$ 时刻, 用不同的仪器测的结果为 z_2 , 误差为 σ_2^2 . 在一些基本假设的前提下, $\bar{\mathbf{x}}(t_2)$ 是一个满足均值为 μ , 方差为 σ^2 的高斯分布, 其中

$$\mu = \frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2} z_1 + \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2} z_2 \quad (4)$$

$$\frac{1}{\sigma^2} = \frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2} \quad (5)$$

根据这个概率密度, $\bar{x}(t_2)$ 的最优估计为 $\bar{x}(t_2) = \mu$ 且其方差为 $\sigma_x^2(t_2) = \sigma^2$. 由公式(2)~(5), $\bar{x}(t_2)$ 和 $\sigma_x^2(t_2)$ 可以重写成如下的形式:

$$\bar{x}(t_2) = \bar{x}(t_1) + K(t_2)[z_2 - \bar{x}(t_1)] \quad (6)$$

$$\sigma_x^2(t_2) = (1 - K(t_2))\sigma_x^2(t_1) \quad (7)$$

$$K(t_2) = \sigma_1^2 / (\sigma_1^2 + \sigma_2^2) \quad (8)$$

式(6)~(8)便是卡尔曼滤波器算法. 从式(4), (5)可以看到, 当 σ_1 等于 σ_2 时, 即两次测量的精度相同, 则正如期望的那样, 船的位置为这两次测量值的简单平均, 另一方面, 当 σ_1 大于 σ_2 时, 测量值 z_1 的不确定性大于测量值 z_2 的不确定性, 因此分配给 z_2 的权值应远大于分配给 z_1 的权值, 而公式(4)正体现了这一点. 同时, 甚至 σ_2 非常大, 估计的误差也小于 σ_1 , 这表明即使获得数据不精确, 它也能够提供一些信息, 从而提高滤波器的精度. 以上是对两个信息源的融合, 而联合卡尔曼滤波^[14]可对多个信息源进行融合, 其所得的最优估计和误差协方差矩阵分别为

$$\bar{x} = (\bar{P}_1^{-1} + \bar{P}_2^{-1} + \dots + \bar{P}_m^{-1})^{-1} \cdot (\bar{P}_1^{-1} \bar{x}_1 + \bar{P}_2^{-1} \bar{x}_2 + \dots + \bar{P}_m^{-1} \bar{x}_m) \quad (9)$$

$$\bar{P} = (\bar{P}_1^{-1} + \bar{P}_2^{-1} + \dots + \bar{P}_m^{-1})^{-1} \quad (10)$$

其中 \bar{x}_i 和 \bar{P}_i 分别为第 i 个卡尔曼滤波器的最优估计及协方差.

2.2 算法描述

由概率论中的乘法定理可得式(11):

$$\begin{aligned} P(\mathbf{x}) &= P(\mathbf{x}, k + 1) \\ &= \prod_{i=1}^n P(x_i | x_1, \dots, x_{i-1}) \\ &= \prod_{i=1}^n P(x_i | \Pi_i) \end{aligned} \quad (11)$$

其中 n 表示变量的维数, $\Pi_i = (x_1, x_2, \dots, x_{i-1})$. 如果用图来表示 $P(\mathbf{x})$, 则可表示成一个有向无环图, 又称为概率图模型, 如图 1 所示. 图中的一个节点代

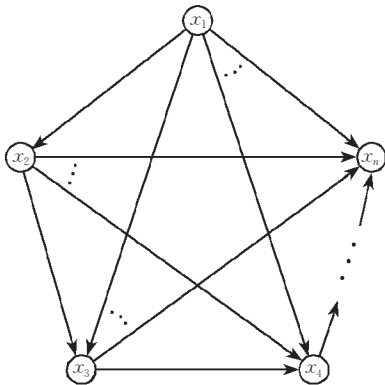


图 1 概率图模型

表一个变量, 两个节点间的边代表变量间的关系, 若节点 x_i 存在一条指向 x_j 的边, 则称 x_i 是 x_j 的父节点. 在式(11)表示的概率图模型中, Π_i 表示节点 x_i 的父节点, 因此一阶分布估计算法是 $|\Pi_i| = 0$ 的图, 二阶分布估计算法是 $|\Pi_i| = 1$ 的图, 而高阶分布估计算法是 $|\Pi_i| \geq 2$ 的图.

假设 n 维变量 \mathbf{x} 的联合概率密度函数为一个多元正态分布:

$$\begin{aligned} P(\mathbf{x}) &\equiv \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \equiv \\ &= (2\pi)^{-\frac{n}{2}} |\boldsymbol{\Sigma}|^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu})\right\} \end{aligned} \quad (12)$$

其中 $\boldsymbol{\mu}$ 是 n 维均值向量, $\boldsymbol{\Sigma}$ 是 $n \times n$ 的协方差矩阵, 而 $|\boldsymbol{\Sigma}|$ 是 $\boldsymbol{\Sigma}$ 的行列式, $\boldsymbol{\Sigma}^{-1}$ 是 $\boldsymbol{\Sigma}$ 的逆矩阵. 则式(11)中的概率密度 $P(\mathbf{x})$ 可以写成每一项独立且服从正态分布的条件概率密度的乘积:

$$\begin{aligned} P(\mathbf{x}) &= \prod_{i=1}^n P(x_i | \Pi_i) \\ &= \prod_{i=1}^n \mathcal{N}\left(\mu_i + \sum_{j=1}^{i-1} a_{ji}(x_j - \mu_j), \sigma_i\right) \end{aligned} \quad (13)$$

这里 μ_i 是变量 x_i 的无条件均值, σ_i 为给定变量 x_1, \dots, x_{i-1} 条件下的方差, 而 a_{ji} 为表示变量 x_j 和 x_i 间关系强度的一个线性系数^[4]. $(\boldsymbol{\mu}, \boldsymbol{\sigma})$ 即为这个概率图模型的参数.

在 MIMIC 中, 给定一个排列 $\boldsymbol{\pi} = (i_1, i_2, \dots, i_n)$, 定义一类概率密度函数 $\mathcal{P}_{\boldsymbol{\pi}}(\mathbf{x})$

$$\begin{aligned} \mathcal{P}_{\boldsymbol{\pi}}(\mathbf{x}) &= \{ \mathcal{P}_{\boldsymbol{\pi}}(\mathbf{x}) | P(\mathbf{x}) = P(x_{i_1} | x_{i_2}) P(x_{i_2} | x_{i_3}) \dots \\ &P(x_{i_{n-1}} | x_{i_n}) P(x_{i_n}) \} \end{aligned} \quad (14)$$

这里 $P(x_{i_n})$ 和 $P(x_{i_j} | x_{i_{j+1}})$, $j = 1, 2, \dots, n-1$ 是服从正态分布的密度函数, 其目标就是选择一个排列 $\boldsymbol{\pi}^*$, 使得 $\mathcal{P}_{\boldsymbol{\pi}^*}(\mathbf{x})$ 与真实密度函数 $P(\mathbf{x})$ 在某种意义上下的误差尽可能小, 这里采用 Kullback-Leibler 散度作为衡量标准, 即

$$\begin{aligned} D_{K-L}(P(\mathbf{x}), \mathcal{P}_{\boldsymbol{\pi}}(\mathbf{x})) &= \\ E_{P(\mathbf{x})} \left\{ \log \frac{P(\mathbf{x})}{\mathcal{P}_{\boldsymbol{\pi}}(\mathbf{x})} \right\} &= \int P(\mathbf{x}) \log \frac{P(\mathbf{x})}{\mathcal{P}_{\boldsymbol{\pi}}(\mathbf{x})} d\mathbf{x} \end{aligned} \quad (15)$$

令 $h(P(\mathbf{x})) = -E_{P(\mathbf{x})}[\log P(\mathbf{x})]$, 有

$$\begin{aligned} D_{K-L}(P(\mathbf{x}), \mathcal{P}_{\boldsymbol{\pi}}(\mathbf{x})) &= \\ -h(P(\mathbf{x})) &+ h(x_{i_1} | x_{i_2}) + h(x_{i_2} | x_{i_3}) + \dots + \\ &h(x_{i_{n-1}} | x_{i_n}) + h(x_{i_n}) \end{aligned} \quad (16)$$

其中 $h(P(\mathbf{x}))$ 与排列 $\boldsymbol{\pi}^*$ 无关, 而 $h(x_{i_1} | x_{i_2}), h(x_{i_2} | x_{i_3}), \dots, h(x_{i_{n-1}} | x_{i_n}), h(x_{i_n})$ 可由 Whittaker 定理^[15] 求出:

Whittaker 定理^[15]. 设 \mathbf{x} 服从 n 维正态分布 $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, 则有

$$h(\mathbf{x}) = \frac{1}{2}n(1 + \log 2\pi) + \frac{1}{2}\log |\Sigma| \quad (17)$$

若将该定理应用到单变量和双变量的正态分布上, 则可得到

$$h(\mathbf{x}) = \frac{1}{2}(1 + \log 2\pi) + \log \sigma_x \quad (18)$$

$$h(\mathbf{x} | y) = \frac{1}{2} \left[(1 + \log 2\pi) + \log \left(\frac{\sigma_x^2 \sigma_y^2 - \sigma_{xy}^2}{\sigma_y^2} \right) \right] \quad (19)$$

其中 σ_x^2 和 σ_y^2 分别为变量 x 和 y 的方差, 而 σ_{xy} 为变量 x 和 y 的协方差. 文献[4]利用贪婪法得到近似最优排列 π^* .

我们将种群随机分成 $(m+1)$ 个子种群, 其大小分别为 $N_i (i=0, 1, \dots, m)$. 子种群 0# 是一个参考种群, 它用卡尔曼滤波器融合出的信息 $\bar{\mathbf{x}}(k-1)$, $\bar{\mathbf{P}}(k-1)$ 来产生下一代种群 Pop_0^k ; 子种群 1# 到 m # 根据估计信息 $(\pi_i^*, \mu_i, \sigma_i) (i=1, 2, \dots, m)$ 按照算法 1 产生种群 Pop_i^k . 进一步, 将 Pop_i^k 随机分成 m 等份, 其每一份与 Pop_i^k 一起选出一部分个体作为第 i 个 MIMIC 的输入数据, MIMIC 在这些数据的基础上建立二阶概率分布模型, 并估计出当前数据的均值 $\tilde{\mathbf{x}}_i(k)$ 和方差 $\tilde{\mathbf{P}}_i(k)$. 最后, 按照式(9), (10)将估计值 $\tilde{\mathbf{x}}_i(k), \tilde{\mathbf{P}}_i(k) (i=1, 2, \dots, m)$ 融合起来, 得到 $\bar{\mathbf{x}}(k), \bar{\mathbf{P}}(k)$. 算法框图见图 2, 详细描述见算法 2.

算法 1. 子种群个体产生算法.

设给子种群 i # 产生个体, $\pi_i^* = (l_1, l_2, \dots, l_n), \mu_i = (\mu_{i,1}, \mu_{i,2}, \dots, \mu_{i,n}), \sigma_i = (\sigma_{i,1}, \sigma_{i,2}, \dots, \sigma_{i,n})$.

1. 以均值为 μ_{i,l_n} , 方差为 σ_{i,l_n} 的正态分布 $\mathcal{N}(\mu_{i,l_n}, \sigma_{i,l_n})$ 产生 x_{l_n} ;
2. 以正态分布 $\mathcal{N}(\mu_{i,l_i} + a_{i+1}l_i (x_{l_{i+1}} - \mu_{i+1}), \sigma_{i,l_i})$ 产

生 $x_{l_i} (i=n-1, n-2, \dots, 1)$, 其中 $a_{i+1}l_i$ 和 σ_{i,l_i} 分别为变量 x_{l_i} 关于变量 $x_{l_{i+1}}$ 的一元回归方程系数和方差.

算法 2. 二阶卡尔曼滤波分布估计算法.

设种群的规模为 N , 将种群随机分成 $(m+1)$ 个子种群, 子种群 i # 包含 $N_i (i=0, 1, \dots, m)$ 个个体, 表示成 Pop_i , 且 $N_0 + N_1 + \dots + N_m = N$. 随机将种群 Pop_0 分成 m 等份, 每份表示为 $\text{Pop}_{0,i} (i=1, \dots, m)$, γ 为选择比例.

1. 随机产生 N 个个体形成初始种群 $\text{Pop}^0, k \leftarrow 0$;
2. 如果终止条件满足, 则输出结果, 停止; 否则转步 3;
3. 从 $\text{Pop}_i^k \cup \text{Pop}_{0,i}^k$ 中选出适应度最高的 $\gamma \cdot (N_i + \frac{N_0}{m})$ 个个体, 第 i 个 MIMIC 根据此数据建立概率分布模型 $(\pi_i^*, \mu_i, \sigma_i)$, 并估计出当前数据的均值 $\tilde{\mathbf{x}}_i(k)$ 和方差 $\tilde{\mathbf{P}}_i(k), i=1, 2, \dots, m$;
4. 按照式(9), (10)将 $\tilde{\mathbf{x}}_i(k), \tilde{\mathbf{P}}_i(k) (i=1, 2, \dots, m)$ 进行融合, 得到 $\bar{\mathbf{x}}(k), \bar{\mathbf{P}}(k)$;
5. 子种群 1# 到 m # 根据算法 1 产生的 $N_i - 1$ 个个体与当前的最优个体一起形成 $\text{Pop}_i^{k+1} (i=1, 2, \dots, m)$; 而子种群 0# 则根据均值为 $\bar{\mathbf{x}}(k)$ 、方差为 $\bar{\mathbf{P}}(k)$ 的正态分布产生 N_0 个个体, 得到种群 Pop_0^{k+1} ; 计算 $\text{Pop}_i^{k+1} (i=0, 1, \dots, m)$ 的适应度;
6. $\text{Pop}^{k+1} \leftarrow \text{Pop}_0^{k+1} \cup \text{Pop}_1^{k+1} \cup \dots \cup \text{Pop}_m^{k+1}, k \leftarrow k+1$, 转步 2.

算法 2 虽然使用了多个子种群, 但它与多种群的进化算法有根本的区别: 多种群的进化算法只是简单地在不同种群间进行个体迁移, 以达到交换信息的目的; 而算法 2 利用卡尔曼滤波信息融合机制对各子种群获得的最优解估计信息进行融合处理, 以减少估计的误差, 因此算法 2 从更深层次利用和交换各子种群的这些信息.

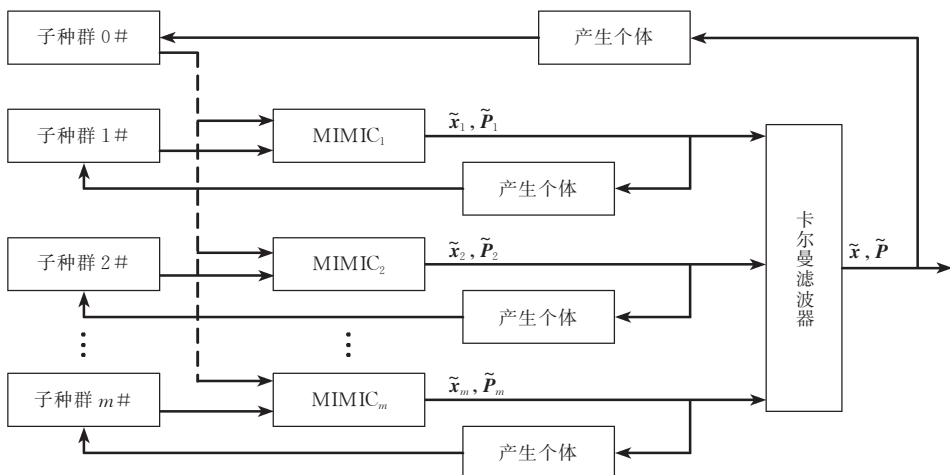


图 2 二阶卡尔曼滤波分布估计算法框图

3 仿真实验

3.1 KFMIMIC 与 MIMIC 的性能比较

为了验证 KFMIMIC 的性能,本文对 3 个标准测试函数 $f_1 \sim f_3$ 进行实验,并与 MIMIC 的结果进行比较. 实验参数设置如下: $n=10, N=1000, m=2, n_0=200, n_1=n_2=400, \gamma=0.2$, 最大运行代数 100, 精度 $\epsilon=10^{-3}$. 停止准则为: 如果 $f_{\min} \neq 0$, 则考察式子 $|f_{\text{best}} - f_{\min}| < \epsilon |f_{\min}|$ 是否成立, 否则考察式子 $|f_{\text{best}}| < \epsilon$, 此处 f_{\min} 和 f_{best} 分别表示全局最优解和到当前代为止找到的最优解. 函数 $f_1 \sim f_3$ 的全局最小值分别为 $-10^5, 0$ 和 0 .

$$f_1(\mathbf{x}) = -1 / (10^{-5} + \sum_{i=1}^n |y_i|);$$

$$y_1 = x_1, y_i = x_i + y_{i-1}, i = 2, \dots, n; -0.16 \leq x_i \leq 0.16,$$

$$f_2(\mathbf{x}) = \sum_{i=1}^n ((x_1 - x_i^2)^2 + (x_i - 1)^2);$$

$$-10 \leq x_i \leq 10,$$

$$f_3(\mathbf{x}) = \frac{1}{4000n} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1;$$

$$-600 \leq x_i \leq 600.$$

表 1 给出了对每个函数进行 100 次实验的结果. 由于函数的变量间都有不同程度的交互作用, MIMIC 容易陷入局部极小值, 尤其是函数 f_2 , MIMIC 在 100 次实验中只有 4 次找到全局最优值. 相反, 本文提出的 KFMIMIC 显示出优越的性能, 在 3 个函数的实验中, 算法 100% 地找到了全局最优值. 这是因为 KFMIMIC 融合多个信息源获得的信息, 并进行信息反馈, 有利于算法跳出局部极值, 从而提高了整个算法的鲁棒性. 同时, KFMIMIC 的优越性能还表现在所需的函数评价次数和运行时间上, 它远远优于 MIMIC 所需的时间代价.

表 1 KFMIMIC 与 MIMIC 的性能比较

		找到全局最优解的实验次数	平均函数评价次数	平均运行时间 (s)
f_1	KFMIMIC	100	71,679	0.831
	MIMIC	75	84,600	1.305
f_2	KFMIMIC	100	56,130	0.561
	MIMIC	4	99,120	1.024
f_3	KFMIMIC	100	26,948	0.354
	MIMIC	80	71,910	0.942

3.2 性能和参数分析

以 f_3 为例, 进一步研究 KFMIMIC 随维数变化和参数 γ (选择比例) 变化的性能, 参数同 3.1 节. 图

3 给出了维数从 10 维增大到 25 维, 算法所需评价次数的增长趋势. 由图 3 可知, 随维数的增大, 算法所需的评价次数增长不大, 通过逼近可知, 算法的复杂度为 $O(n^2)$.

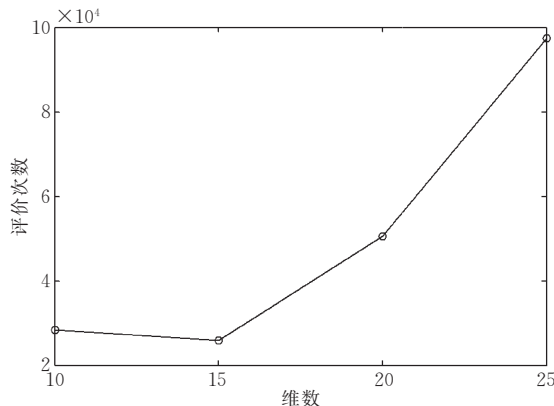


图 3 KFMIMIC 随维数变化的性能

图 4 给出了参数 γ 在 $[0.1, 0.8]$ 变化, KFMIMIC 所需评价次数的变化趋势图. 从图 4 可知, 评价次数随参数 γ 的增大呈先变小后变大的趋势. 这是因为由于 γ 很小, 只选出很小一部分较好的个体, 这时算法的收敛速度较快, 但容易早熟, 而当参数 γ 过大, 则建立的概率分布模型没有突出适应度较高的个体, 算法的收敛速度较慢, 需要更多的评价次数, 因此参数 γ 一般可在 $[0.2, 0.5]$ 内选取.

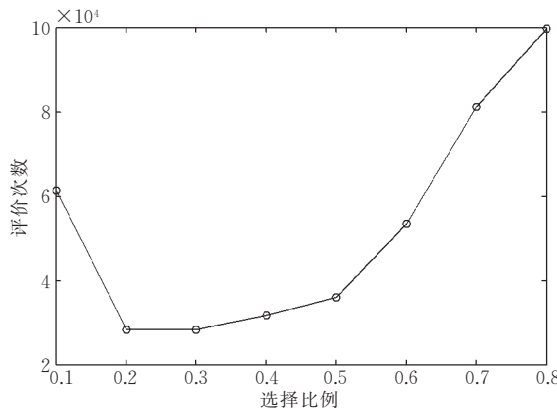


图 4 KFMIMIC 随选择比例变化的性能

4 结 论

从卡尔曼滤波这一全新的角度分析了分布估计算法, 并针对二阶分布估计算法容易陷入局部极值的缺点, 提出了二阶卡尔曼滤波分布估计算法. 通过对比实验, 结果表明本文算法的稳定性有了很大程度的提高, 且其全局搜索能力得到很大的增强. 同时通过算法的性能和参数分析, 得到其复杂度 $O(n^2)$

和参数 γ 的选取区间 $[0.2, 0.5]$. 如何将这种基于卡尔曼滤波的方式推广到高阶分布估计算法上, 将是我們下一步的工作.

参 考 文 献

- 1 Baluja S. . Population-based incremental learning: A method for integrating genetic searching based function optimization and competing learning. Carnegie Mellon University, Pittsburgh, PA, USA; Technical Report CMU-CS-94-163, 1994
- 2 Muhlenbein H. . The equation for response to selection and its use for prediction. *Evolutionary Computation*, 1998, 5(3): 303~346
- 3 De Bonet, Isbell J. S. , Viola P. . MIMIC: Finding optima by estimating probability density. In: Mozer M. C. , Jordan M. I. , Petsche T. eds. . *Advances in Neural Information Processing System*. Cambridge: The MIT Press, 1997, 424~431
- 4 Larranga P. , Etzeberria R. , Lozano A. , Pefia J. M. . Optimization by learning and simulation of Bayesian and Gaussian networks. In: *Proceedings of the 2000 Genetic and Evolutionary Computation Conference Workshop Program*, Las Vegas, Nevada, USA, 2000, 201~204
- 5 Baluja S. , Davies S. . Using optimal dependency-tree for combinatorial optimization: Learning the structure of the search space. Carnegie Mellon University, Pittsburgh, PA, USA; Technical Report CMU-CS-97-107, 1997
- 6 Soto M. , Ochoa A. , Acid S. *et al.* . Introduction the polytree approximation of distribution algorithm. In: *Proceedings of the 2nd Symposium on Artificial Intelligent Adaptive Systems, CIMAF'99*, La Habana, 1999, 360~367
- 7 Peliken M. , Muhlenbein H. . The bivariate marginal distribution algorithm. In: Roy R. , Furnhashi T. , Chandhery P. K. eds. . *Advance in Soft Computing Engineering Design and Manufacturing*. London: Springer-Verlag, 1999, 521~535
- 8 Muhlenbein H. , Mahnig T. . FDA-A scalable evolutionary algorithm for the optimization of additively decomposed function. *Evolutionary Computation*, 1999, 7(4): 353~376
- 9 Pelikan M. , Goldberg D. E. , Cantu-Paz E. . BOA: The Bayesian optimization algorithm. In: *Proceedings of the Genetic and Evolutionary Computation Conference*, Orlando, Florida, USA, 1999, 525~532
- 10 Zhang B. T. . A Bayesian framework for evolutionary computation. In: *Proceedings of the 1999 Congress on Evolutionary Computation*, 1999, 1: 722~728
- 11 Bosman P. A. N. , Thierens D. . Expanding from discrete to continuous estimation of distribution algorithm: The IDEA. In: Schoenauer M. et al. eds. . *Parallel Problem Solving from Nature VI*, LNCS 1917, Berlin: Springer, 2000, 767~776
- 12 Lin Y-P. A survey of evolutionary algorithm based on probabilistic model. *Journal of Computer Research and Development*, 2001, 38(1): 43~49(in Chinese)
(林亚平. 概率分析进化算法及其研究进展. *计算机研究与发展*, 2001, 38(1): 43~49)
- 13 Zhang Q. F. , Muhlenbein. On convergence of a class of optimization algorithms using estimation of distribution. *IEEE Transactions on Evolutionary Computation*, 2004, 8(2): 127~136
- 14 Carlson N. A. . Information-sharing approach to federated Kalman filtering. In: *Proceedings of National Aerospace Electronics Conference*, Dayton, OH, USA, 1998, 230~235
- 15 Whittaker J. . *Graphical Models in Applied Multivariate Statistics*. Chichester, England: John Wiley and Sons, 1990



ZHONG Wei-Cai, born in 1977, Ph. D. candidate. His current research interests include evolutionary computation, data mining, multi-media applications, and pattern recognition.

LIU Jing, born in 1977, Ph. D. candidate. Her current

research interests include evolutionary computation, data mining, video compression, and machine learning.

LIU Fang, born in 1963, M. S. , professor. Her current research interests include artificial intelligence, pattern recognition, and evolutionary computation.

JIAO Li-Cheng, born in 1959, Ph. D. , professor and Ph. D. supervisor. His current research interests include evolutionary computation, neural network, and wavelet theory.

Background

This work is supported by the National Natural Science Foundation of China under grant No. 60133010, and the title is the Theories and Applications of Evolutionary Computation. They have made researches on many fields of the evolutionary computation, such as immune evolutionary algo-

gorithms, quantum evolutionary algorithms, coevolutionary algorithms and so on. This paper belongs to the part of coevolutionary computation and focuses on proposing a new coevolutionary mechanism.