

文章编号: 1673-5196(2020)01-0106-05

基于 GDBA 算法目标跟踪的粒子多样性研究

杜先君^{1,2,3}, 马金斗¹

(1. 兰州理工大学 电气工程与信息工程学院, 甘肃 兰州 730050; 2. 兰州理工大学 甘肃省工业过程先进控制重点实验室, 甘肃 兰州 730050; 3. 兰州理工大学 电气与控制工程国家实验教学示范中心, 甘肃 兰州 730050)

摘要: 针对传统目标跟踪算法搜索范围小、跟踪精度低的缺点, 提出一种基于遗传扰动机制的改进蝙蝠算法(GD-BA), 该算法引入了遗传竞争机制, 根据优化的优劣情况调整遗传算法的交叉率和变异率, 使得种群具有遗传性和变异性, 同时扩大了搜索范围, 提高了粒子多样性, 改善了跟踪精度。

关键词: 竞争机制; 跟踪精度; GDBA 算法; 粒子多样性

中图分类号: TP301.6 **文献标志码:** A

Investigation of particle variety based on target tracking of GDBA algorithm

DU Xian-jun^{1,2,3}, MA Jin-dou¹

(1. College of Electrical and Information Engineering, Lanzhou Univ. of Tech., Lanzhou 730050, China; 2. Key Laboratory of Gansu Advanced Control for Industrial Process, Lanzhou Univ. of Tech., Lanzhou 730050, China; 3. National Demonstration Center for Experimental Electrical and Control Engineering Education, Lanzhou 730050, China)

Abstract: Aimed at the defect of small searching scope and low tracking accuracy of traditional target tracking algorithm, an improved bat algorithm(GDBA) is proposed based on genetic disturbance mechanism. In this algorithm, the genetic competitive mechanism is introduced to improve the bat algorithm, the crossover factor and the mutation rate in the genetic algorithm are adjusted according to the good-bad condition of optimization, so that the population will be made to have heritability and diversity and meantime, the searching range will be expanded and the tracking accuracy improved.

Key words: competition mechanism; tracking accuracy; GDBA algorithm; particle variety

目标跟踪研究是当今计算机研究领域的热点, 它的应用前景广阔, 现在已经广泛地应用在视频分析^[1]、智能控制^[2]、飞行器导航^[3]等多个不同的领域。李雅倩等^[4]提出了一种能适应尺度变化、姿态变化以及形状畸变的鱼眼视频目标跟踪方法, 用来解决畸变严重的鱼眼图像中的目标跟踪。段建民等^[5]提出了一种改进智能汽车目标跟踪的方法, 用来改善自动驾驶性能。吴俊辰等^[6]提出了一种基于随机集的点目标和扩展目标联合跟踪算法, 用来解决雷达对飞机、船舶和导弹等机动军事目标精确探测的鲁棒性问题。

目前对目标跟踪的研究主要集中在用智能优化算法改进传统目标跟踪的领域, 并且随着人们对智能化的要求越来越高, 多个智能优化算法相互结合

从而弥补相互之间缺点的研究越来越多。滕飞等^[7]提出了一种基于 Student's t 分布的自适应重采样粒子滤波算法, 用来提高粒子的多样性。杜正聪等^[8]提出了应用遗传算法的进化思想优化粒子滤波重采样过程, 用来解决粒子滤波权值退化问题。孟凡琨等^[9]提出了一种基于蝙蝠算法的粒子滤波算法, 利用蝙蝠算法优化粒子滤波算法, 解决标准粒子滤波算法容易出现的粒子贫化问题。也有很多学者在智能优化算法改进目标跟踪的领域作了大量的研究。马兆南等^[10]提出了一种改进的粒子滤波目标跟踪算法, 用来解决传统目标跟踪定位时间长、定位精度低的缺点。刘一杨等^[11]提出了一种用遗传算法优化机器人运动轨迹跟踪的算法, 用来提高机器人跟踪控制的精确性和稳定性。

但在实际使用的过程中, 以上改进的目标跟踪算法因背景干扰多、跟踪的目标移动多变等因素影响, 使得搜索范围减小、跟踪精度降低, 从而降低了

收稿日期: 2018-07-02

作者简介: 杜先君(1979-), 男, 浙江杭州人, 博士, 副教授。

目标的跟踪性能.

本文主要针对以上目标跟踪的缺点,提出一种基于遗传扰动机制的改进蝙蝠优化算法(GDBA).该算法在基本蝙蝠算法的基础上,引入遗传变异机制,自适应地调整交叉率和变异率,增加种群的多样性,从而扩大搜索范围、提高跟踪精度.

1 基本蝙蝠算法

BA 算法^[12]是利用蝙蝠在觅食时所发出脉冲的频率 f 、响度 A 和发射率 r 的变化而建立的模型.

1) 在算法迭代过程中,蝙蝠个体 i 的参数更新可描述为

$$f_i = f_{\min} + (f_{\max} - f_{\min})\beta \quad (1)$$

$$v_i(k) = v_i(k-1) + (x_i(k) - x^*(k))f_i \quad (2)$$

$$x_i(k) = x_i(k-1) + v_i(k) \quad (3)$$

其中: k 为当前迭代次数; $\beta \in [0, 1]$ 为一个随机向量,服从均匀分布; $x_i(k)$ 为蝙蝠个体 i 当前的位置; $x^*(k)$ 为当前的全局最优位置; f_i 为蝙蝠个体 i 当前发出的脉冲频率; $v_i(k)$ 为当前蝙蝠个体 i 的飞行速度.

2) 在当前全局最优位置附近进行局部搜索:

$$x'_i(k) = x^*(k) + \epsilon A(k) \quad (4)$$

其中: $\epsilon \in [-1, 1]$ 为一个随机数; $A(k)$ 为迭代次数为 k 时的蝙蝠个体的平均音强.

3) 更新脉冲响度 A 和脉冲发射频率 r :

$$r(k+1) = r_{\max} [1 - \exp(-\gamma k)] \quad (5)$$

$$A(k+1) = \alpha A(k) \quad (6)$$

其中: α 和 γ 为常量,一般取值为 0.9,并且随着迭代次数的增加,可以得到

$$k \rightarrow \infty, \quad \begin{cases} A(k) \rightarrow 0 \\ r(k) \rightarrow r_{\max} \end{cases}$$

2 本文的 GDBA 算法

本文提出一种基于遗传扰动机制的改进蝙蝠优化算法(GDBA),该算法在种群局部搜索时,应用遗传算法具有的交叉特性和变异特性产生子代,并且从中选择适应度最优的子代,再将其和蝙蝠算法中与之相对应的子代进行对比、竞争,选择适应度更好的子代作为最优子代,从而使得改进的算法在后期具有更加丰富的种群多样性以及更加广阔的搜索范围.

GDBA 算法具体实现步骤如下:

第 1 步,参数初始化:初始化搜索种群规模 n ,与搜索种群规模相对应的脉冲频率范围 $[f_{\min}, f_{\max}]$,最大脉冲响度 A_{\max} ,最大脉冲频率 r_{\max} ,迭代

次数 $Niter$,交叉概率 p_c ,变异概率 p_m ,数据范围 $bound$,遗传算法的进化代数 $maxgen$,精度 $tol = 10^{-5}$.

第 2 步,初始化蝙蝠的位置 X_i :随机生成 n 只蝙蝠,并设第 i 只蝙蝠为 $X_i = (x_{i,1}, x_{i,2}, \dots, x_{i,D})$, $i \in (1, 2, 3, \dots, n)$, D 为维数,根据式(7)计算出蝙蝠的适应度值 $fit(X_i)$,然后选取全局最优适应度值 fit_{\max} ,从中寻找出最优蝙蝠个体 x^* :

$$fit = \exp\left[-\frac{1}{2R_k}(z_{\text{new}} - z_{\text{pred}}(i))\right] \quad (7)$$

其中: R_k 为测量噪声方差; z_{new} 为滤波器最新的观测值; z_{pred} 为滤波器预测的观测值.

第 3 步,更新蝙蝠的空间位置:根据式(1)更新蝙蝠发射信号的脉冲频率,根据式(2)更新蝙蝠的飞行速度 v_i ,根据式(3)更新蝙蝠的空间位置 X_i .

生成随机数 $rand_1$,若 $rand_1 > r_{Niter}$,则按照式(4)更新蝙蝠位置,并根据式(5,6)进行频度和音强指标的更新.

第 4 步,引入遗传扰动机制:

1) 将当前更新后的蝙蝠种群作为遗传算法初始种群,并计算种群个体的适应度值,记为 $fit_1(X_i)$,并计算 $fit_1(X_i)$ 与全局最优适应度 fit_{\max} 的差值 Δd_i ,根据 Δd_i 的大小调整下一代蝙蝠个体的交叉概率 p_c 和变异概率 p_m ,具体调整策略如下:

(1) 若当代蝙蝠个体的适应度值与全局最优适应度 fit_{\max} 的差值 Δd_i 较大时,为保持子代的稳定性,则交叉概率 p_c 应该向较大的数值调整,变异概率 p_m 相对应地应该向较小的数值调整;

(2) 若 Δd_i 较小时,为保持子代的多样性,避免陷入早熟,则交叉率 p_c 就应该向较小的数值调整,变异概率 p_m 相对应地应该向较大的数值调整;

(3) 当 Δd_i 达到优化精度要求时,交叉率 p_c 和变异率 p_m 应该保持数值不变.

2) 按照调整后的交叉率和变异率分别完成选择、交叉、变异过程,产生子代,再将子代代入目标函数,计算出它们的适应度值,记为 $fit_2(X_i)$.

3) 比较适应度值 $fit_1(X_i)$ 和 $fit_2(X_i)$,选取最大的适应度值并记为 $fit_{\text{new}}(X_i)$,将最大适应度值 $fit_{\text{new}}(X_i)$ 所对应的蝙蝠位置 X_i 作为蝙蝠更新位置,并记录.

第 5 步,再生成随机数 $rand_2$,若 $rand_2 < A_{Niter}$,且适应度值有所变化时,则认为引入的遗传扰动机制产生作用,故而将第 4 步蝙蝠更新的位置作为蝙蝠的位置,且若 $fit_{\text{new}}(X_i) \geq f_{\max}$,则更新全局最优值 x^* 和全局最优适应度值 fit_{\max} .

第 6 步,根据模型的不同作出相应的输出.
本文算法的整体流程如图 1 所示.

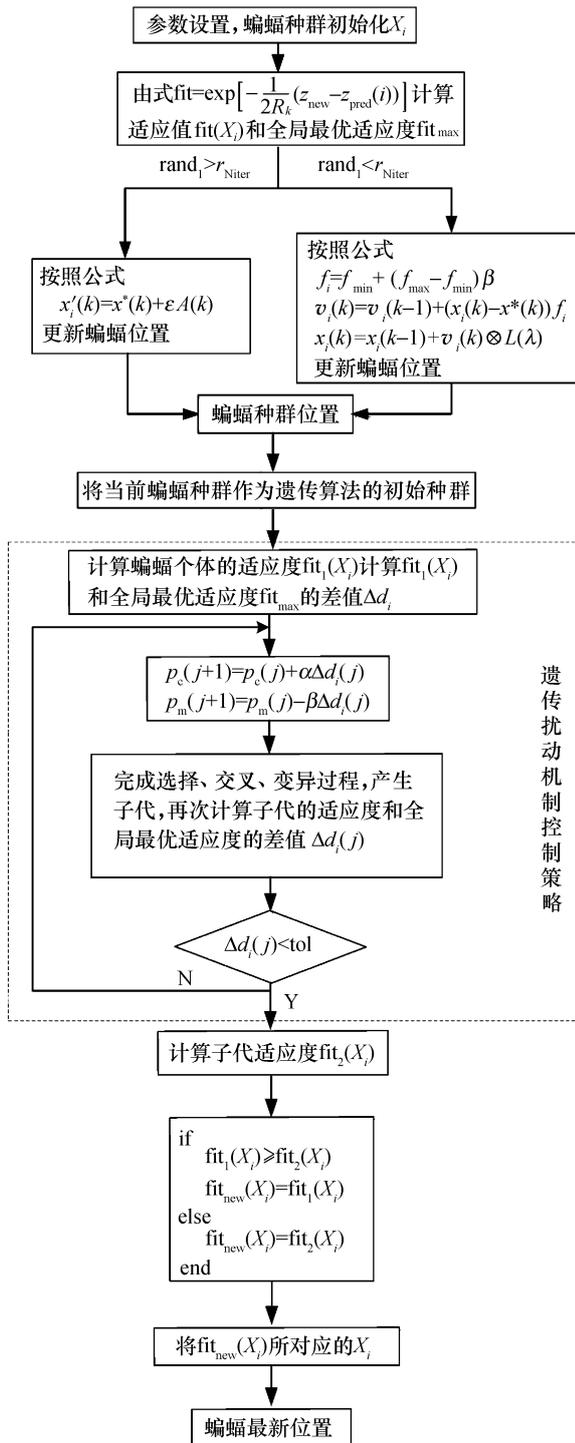


图 1 GDBA 算法的控制流程图

Fig.1 Control flow-chart of GDBA algorithm

3 实验结果与分析

3.1 非平稳经济学估计问题

选取单变量非静态增长模型(UNGM 模型)为目标跟踪对象,仿真的过程模型和状态模型如下.

1) 系统过程模型:

$$x(t) = 0.5x(t-1) + \frac{25x(t-1)}{1 + [x(t-1)]^2} + 0.8\cos[1.2(t-1)] + w(t) \quad (8)$$

2) 状态测量模型:

$$z(t) = x(t)^2/20 + v(t) \quad (9)$$

式(8,9)中: $w(t)$ 和 $v(t)$ 为零均值高斯噪声,设系统噪声 $w(t)$ 的方差为 $Q=10$,测量噪声 $v(t)$ 的方差为 $R=1$.仿真步长 Time 设置为 60,初始化蝙蝠种群数量为 $n=50$,最大响度为 $A_{max}=0.2$,最大脉冲响度 $r_{max}=0.7$,遗传算法的迭代次数 $maxgen=10$,脉冲频率 $f_{min}=0, f_{max}=100$ kHz,种群数量 $sizepop=10$,交叉率 $p_c(1)=0.6, p_m(1)=0.01, p_c = p_c + 0.005\Delta d, p_m = p_m - 0.005\Delta d, \Delta d$ 为当前子代适应度值与全局最优适应度的差值,数据范围 $bound = [-2, 2]$,算法最终迭代次数 $Niter=100$.以单变量非静态增长模型(UNGM 模型)为目标跟踪对象,用杜正聪等^[8]提出的遗传算法优化粒子滤波算法、孟凡琨等^[9]提出的蝙蝠算法优化粒子滤波的算法、马兆南等^[10]提出的改进的粒子滤波算法和本文提出的算法对其进行跟踪,实验参数见表 1.

表 1 实验参数

Tab.1 List of experimental parameters

算法	n	f_{min}/kHz	f_{max}/kHz	Niter
文献[8]	50			100
文献[9]	50	0	100	100
文献[10]	50			100
本文算法	50	0	100	100

均方根误差计算公式为

$$RMSE = \left[\frac{1}{T} \sum_{t=1}^T (x_t - \hat{x}_t)^2 \right]^{\frac{1}{2}} \quad (10)$$

1) 当粒子数 $N=20, Q=10$ 时,仿真结果如图 2 和图 3 所示.

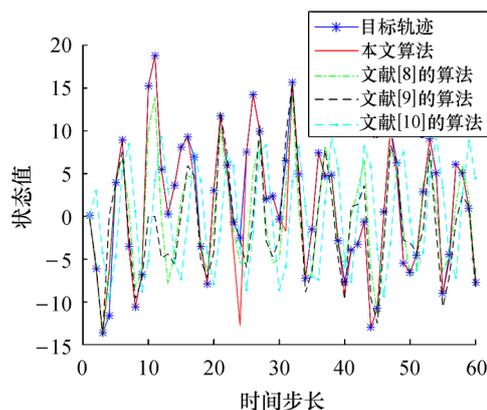


图 2 目标跟踪($N=20, Q=10$)

Fig.2 Target tracking($N=20, Q=10$)

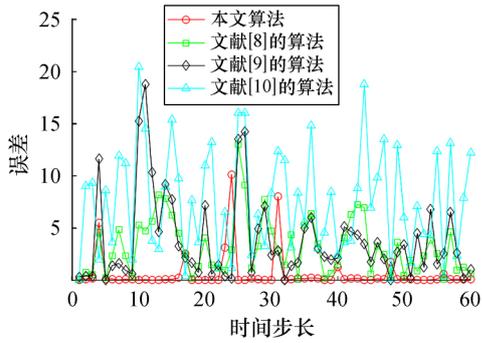


图 3 目标跟踪的均方误差 ($N=20, Q=10$)

Fig. 3 Mean square error of target tracking ($N=20, Q=10$)

2) 当粒子数 $N=100, Q=10$ 时, 仿真结果如图 4 和图 5 所示.

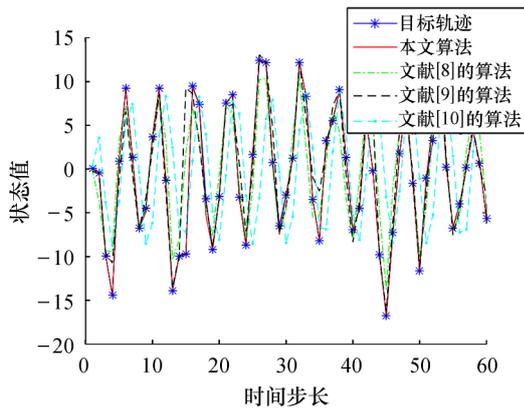


图 4 目标跟踪 ($N=100, Q=10$)

Fig. 4 Target tracking ($N=100, Q=10$)

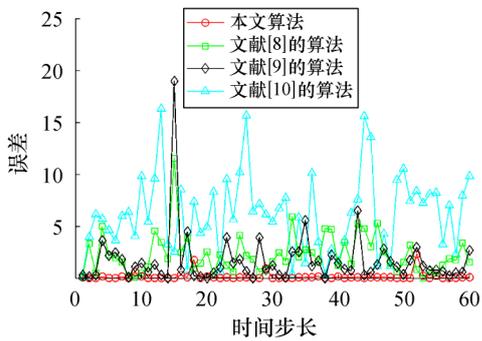


图 5 目标跟踪的均方误差 ($N=100, Q=10$)

Fig. 5 Mean square error of target tracking ($N=100, Q=10$)

由图 2 ~ 5 可知, 随着优化粒子数的增加, 四种优化算法的跟踪效果都有明显的改善, 但是本文算法的目标跟踪精度更高, 跟踪误差更小.

3.2 粒子多样性测试

以优化粒子数 $N=20$, 噪声方差 $Q=10$ 的系统为研究对象, 选取 $k=10, k=25, k=50$ 的粒子分布状态进行实验, 仿真结果如图 6 ~ 8 所示.

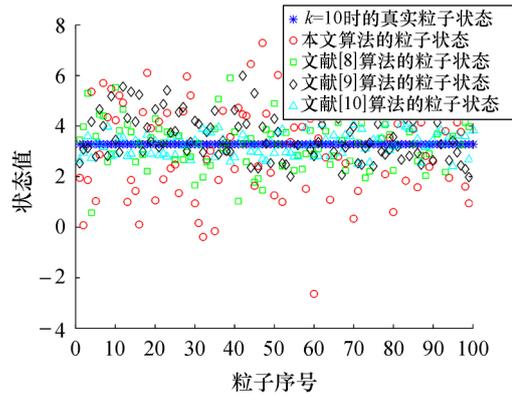


图 6 $k=10$ 时的粒子状态分布图

Fig. 6 Particle state distribution map when $k=10$

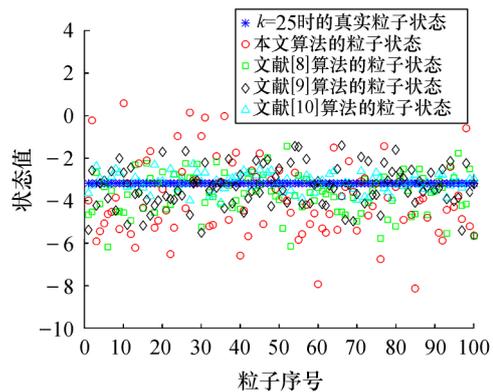


图 7 $k=25$ 时的粒子状态分布图

Fig. 7 Particle state distribution map when $k=25$

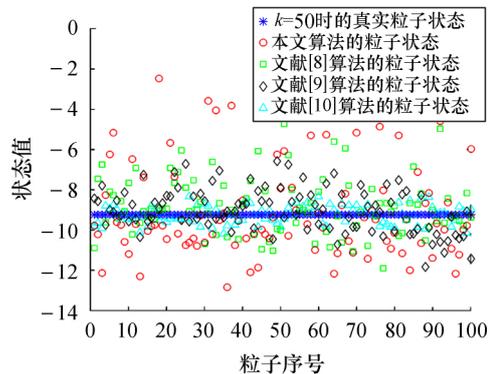


图 8 $k=50$ 时的粒子状态分布图

Fig. 8 Particle state distribution map when $k=50$

从图 6 ~ 8 可以看出, 改进的粒子滤波算法、遗传算法优化粒子滤波算法、蝙蝠算法优化粒子滤波算法在粒子优化迭代的过程中, 对粒子的多样性都有所改善, 但本文算法的粒子分布多样性更加丰富, 为避免早熟问题、增强全局搜索能力、解决粒子贫化问题、提高跟踪精度提供了坚实基础.

4 结语

本文提出了一种基于遗传扰动机制的改进蝙蝠算法的目标跟踪方法,通过引入遗传变异机制,自适应地调整交叉率和变异率,从而增加了种群的多样性,增强了全局搜索能力,提高了跟踪精度.选取非平稳经济学估计问题作为实验对象,对改进的粒子滤波算法、遗传算法优化粒子滤波算法、蝙蝠算法优化粒子滤波算法和本文算法进行实验,实验结果表明本文提出的方法跟踪能力更强,跟踪精度更高.并选取 $k=10$ 、 $k=25$ 、 $k=50$ 的粒子状态分布进行测试,结果表明本文算法的粒子多样性更加丰富.

参考文献:

- [1] 管 皓,薛向阳,安志勇.深度学习在视频目标跟踪中的应用进展与展望 [J].自动化学报,2016,42(6):834-847.
- [2] 李朕阳,郎 朗,陈孟元.基于SR-CKF的机器人动态目标跟踪算法 [J].电子测量与仪器学报,2016,30(8):1198-1205.
- [3] 张天翼,杨 忠,韩家明,等.基于连续自适应均值漂移和立体视觉的无人机目标跟踪方法 [J].应用科技,2018,45(2):55-59.
- [4] 李雅倩,贾 璐,李海滨,等.基于压缩特征的鱼眼视频目标跟踪算法研究 [J].电子与信息学报,2018,40(5):1242-1249.
- [5] 段建民,田晓生,夏 天,等.基于模型预测控制的智能汽车目标路径跟踪方法研究 [J].汽车技术,2017(8):6-11.
- [6] 吴俊辰,周 峰.点目标和扩展目标联合跟踪算法 [J].计算机应用研究,2017,34(9):2586-2589.
- [7] 滕 飞,薛 磊,李修和.基于 Student's t 分布的自适应重采样粒子滤波算法 [J].控制与决策,2018,33(2):361-365.
- [8] 杜正聪,邓 寻.基于自适应遗传算法的粒子滤波器 [J].成都理工大学学报(自然科学版),2017,44(5):636-640.
- [9] 孟凡琨,巨永锋,温立民,等.基于APRBA粒子滤波算法的移动目标跟踪算法 [J].控制工程,2017,24(6):1212-1217.
- [10] 马兆南,裴腾达,张 浩.一种改进的粒子滤波目标跟踪算法 [J].辽宁工程技术大学学报(自然科学版),2016,35(9):978-982.
- [11] 刘一扬,刘明明.基于改进遗传算法的机器人运动轨迹跟踪控制 [J].机械传动,2016,40(3):39-42.
- [12] 吕石磊,黄永霖,陈海强,等.基于自适应步长的改进蝙蝠算法 [J].控制与决策,2018,33(3):557-564.