

DOI:10.16515/j.cnki.32-1722/n.2018.02.0005

一种集成学习的差分进化算法

谢 宇^{1,2}, 王庆龙^{1*}, 赵春霞^{2*}

(1. 合肥学院电子信息与电气工程系, 安徽 合肥 230601; 2. 南京理工大学计算机科学与工程学院, 江苏 南京 210094)

摘 要: 提出一种使用种群规模线性调节机制的集成学习差分进化算法。该算法在每次迭代后根据适应度值对种群中的个体进行排序, 剔除适应度最差的个体来改变种群规模(population size, NP)。这使得个体的数量随着评估次数的增加而线性减小, 有效提高了收敛性。将该算法应用到测试函数集以及进行特征子集选择时获得了优异的平均分类正确率。

关键词: 集成学习; 差分进化; 进化计算; 种群规模

中图分类号: TP301

文献标识码: A

文章编号: 1672-755X(2018)02-0019-04

Differential Evolution Algorithm Based on Integrated Learning

XIE Yu^{1,2}, WANG Qing-long^{1*}, ZHAO Chun-xia^{2*}

(1. Hefei University, Hefei 230601, China; 2. Nanjing University of Science and Technology, Nanjing 210094, China)

Abstract: A differential evolution (DE) algorithm based on intergrated learning is proposed in this paper, which uses population scale linear regulation mechanism. After each iteration, the individuals in the population are sorted according to the fitness value, and the worst fitness individuals are eliminated to change the population size (NP). This makes the number of individuals decrease linearly with the increase of the number of evaluations, which effectively improves the convergence. The algorithm is applied to test function set and feature subset selection, and the average classification accuracy is obtained.

Key words: integrated learning; differential evolution; evolutionary computation; population size

集成学习(Ensemble Learning)是机器学习的一个重要的研究方向,它通过某种方式将一些简单的分类器组合成一个集成学习机来解决同一个问题,以期取得比单个分类器更好的性能^[1]。有研究者使用一组神经网络的组合来解决他们遇到的问题,实验结果表明使用这一组神经网络形成的集成,比最好的个体神经网络的性能还要好^[1]。Wang 等^[2]在进化计算领域也使用了类似的集成方法,该方法叫做组合试验向量产生策略和控制参数的差分进化算法(CoDE),算法对多个策略和多个参数组合进行了集成。在 CoDE 算法中使用的三种 DE 策略分别是:DE/rand/1/bin、DE/rand/2/bin 和 DE/current-to-rand/1/bin。而对于尺度因子和变异概率因子,这里选择的三种组合分别为适合处理可分解问题的 $[F=1.0, Cr=0.1]$ 组合,可以保持种群多样性以及适合全局搜索的 $[F=1.0, Cr=0.9]$ 组合,以及可以加速种群收敛的 $[F=0.8, Cr=0.2]$ 组合。在 CoDE 算法中,每个策略都从参数池里随机选择一组参数,用三种不同的策略分

收稿日期: 2018-06-02

基金项目: 国家自然科学基金面上项目(61773210);国家自然科学基金面上项目(41775025)

作者简介: 谢宇(1977—),男,安徽芜湖人,讲师,博士,主要从事模式识别和进化计算研究。

通信作者: 王庆龙(1968—),男,安徽六安人,教授,博士,主要从事过程控制和新能源领域研究。

赵春霞(1964—),女,北京人,教授,博士,主要从事模式识别和智能机器人领域研究。

别给目标向量产生一个试验向量,根据产生的试验向量的适应度值来选择一个个体的进入下一次迭代。

1 LCoDE 算法

CoDE 算法的种群规模是保持不变的,而且使用的迭代终止条件是适应度评估次数达到预先设定的值。在进化的后期,种群会集中在全局最优解附近,在使用多个策略和多种参数组合的情况下,过多的个体对于找到全局最优解并没有太大的帮助。为此作者提出使用种群规模线性调节机制的 CoDE 算法(LCoDE)来提高算法的收敛性能^[1]。

$$NP = NP_{\max} - \text{round}\left(\frac{(NP_{\max} - NP_{\min})FEs}{MAXFEs}\right) \quad (1)$$

其中 NP_{\max} 为初始种群规模, NP_{\min} 是最小种群规模, FEs 为当前适应度评估次数, $MAXFEs$ 为最大评估次数, $\text{round}(\cdot)$ 为就近取整函数。

根据该种群规模调节机制, LCoDE 在每次迭代后根据适应度值对种群中的个体进行排序, 当种群规模 NP 发生改变时, 剔除适应度最差的个体。这样种群中包含的个体的数量将随着评估次数的增加而线性减小。

2 LCoDE 算法性能分析

在对比实验中, CoDE 使用的参数是, 染色体维数 $D=30$, 种群规模 $NP=30$, 迭代终止条件为对适应度函数的评估次数达到 $D \times 10\,000$ 。LCoDE 使用的参数分别为染色体维数 $D=30$, $NP_{\max}=30$, $NP_{\min}=6$, $MAXFEs$ 等于 $D \times 10\,000$ 。对于每个测试函数, 两种算法分别运行 25 次, 取每次运行的最佳适应度的均值作为两种算法的对比标准。对于 CEC 2005 优化函数测试集的前 14 个测试函数的测试结果如表 1 所示。LCoDE 在其中的 9 个函数上的结果优于 CoDE 算法; 两种算法在函数 F_1 和 F_9 上的结果相当, 都收敛到全局最优; 在其它 3 个函数上 LCoDE 的结果略逊于 CoDE。表明 LCoDE 算法有更大的优势。

表 1 CoDE 和 LCoDE 的结果对比

测试函数	CoDE		LCoDE		
	最佳结果均值	标准差	最佳结果均值	标准差	
单峰函数	F_1	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	F_2	1.69E-15	3.95E-15	1.72E-19	2.02E-19
	F_3	1.05E+05	6.25E+04	8.37E+04	2.22E+04
	F_4	5.81E-03	1.38E-02	4.97E-02	5.12E-02
	F_5	3.31E+02	3.44E+02	6.05E+02	4.12E+02
基本多峰函数	F_6	1.60E-01	7.85E-01	9.35E-04	1.25E-03
	F_7	7.46E-03	8.55E-03	2.91E-01	2.22E-01
	F_8	2.01E+01	1.41E-01	2.01E+01	8.85E-02
	F_9	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	F_{10}	4.15E+01	1.16E+01	3.38E+01	4.98E+00
	F_{11}	1.18E+01	3.40E+00	1.04E+01	1.43E+00
	F_{12}	3.05E+03	3.80E+03	2.08E+03	2.47E+03
扩展多峰函数	F_{13}	1.57E+00	3.27E-01	1.54E+00	2.73E-01
	F_{14}	1.23E+01	4.81E-01	1.23E+01	2.53E-01

注:加粗字体表示算法的结果较优,下同。

除了比较两种算法的寻优结果,这里还比较了两种算法的收敛速度,结果如图 1 所示。

LCoDE 还和一些经过实验验证且非常优秀的改进型差分进化算法如 JADE^[3]、jDE^[4]、SaDE^[5] 和 EPSDE^[6] 等进行了对比(表 2)。结果表明 LCoDE 算法在收敛结果和收敛速度上均占优。

在 JADE 算法进化的每一代中,每个个体根据正态分布随机产生各自的交叉概率因子,而尺度因子 F 来自柯西分布随机数。jDE 算法把控制参数 F 和 Cr 加入到每个个体的编码中,并通过两个新的参数 τ_1 和 τ_2 来调整它们的。SaDE 算法中控制参数的取值和试验向量产生策略都是根据产生过合适解的经验逐渐地进行自调整的。EPSDE 算法也类似于一种集成多种策略和参数的算法,它包含不同向量产生策略的

表 2 JADE、jDE、SaDE、EPSDE 和 LCoDE 的测试结果对比(其它算法的数据来自文献)

测试函数	JADE		jDE		SaDE		EPSDE		LCoDE	
	最佳结果均值	标准差	最佳结果均值	标准差	最佳结果均值	标准差	最佳结果均值	标准差	最佳结果均值	标准差
F ₁	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F ₂	1.07E-28	1.00E-28	1.11E-06	1.96E-06	8.26E-06	1.65E-05	4.23E-26	4.07E-26	1.72E-19	2.02E-19
F ₃	8.42E+03	7.26E+03	1.98E+05	1.10E+05	4.27E+05	2.08E+05	8.74E+05	3.28E+06	8.37E+04	2.22E+04
F ₄	1.73E-16	5.43E-16	4.40E-02	1.26E-01	1.77E+02	2.67E+02	3.49E+02	2.23E+03	4.97E-02	5.12E-02
F ₅	8.59E-08	5.23E-07	5.11E+02	4.40E+02	3.25E+03	5.90E+02	1.40E+03	7.12E+02	6.05E+02	4.12E+02
F ₆	1.02E+01	2.96E+01	2.35E+01	2.50E+01	5.31E+01	3.25E+01	6.38E-01	1.49E+00	9.35E-04	1.25E-03
F ₇	8.07E-03	7.42E-03	1.18E-02	7.78E-03	1.57E-02	1.38E-02	1.77E-02	1.34E-02	2.91E-01	2.22E-01
F ₈	2.09E+01	1.68E-01	2.09E+01	4.86E-02	2.09E+01	4.95E-02	2.09E+01	5.81E-02	2.01E+01	8.85E-02
F ₉	0.00E+00	0.00E+00	0.00E+00	0.00E+00	2.39E-01	4.33E-01	3.98E-02	1.99E-01	0.00E+00	0.00E+00
F ₁₀	2.41E+01	4.61E+00	5.54E+01	8.46E+00	4.72E+01	1.01E+01	5.36E+01	3.03E+01	3.38E+01	4.98E+00
F ₁₁	2.53E+01	1.65E+00	2.79E+01	1.61E+00	1.65E+01	2.42E+00	3.56E+01	3.88E+00	1.04E+01	1.43E+00
F ₁₂	6.15E+03	4.79E+03	8.63E+03	8.31E+03	3.02E+03	2.33E+03	3.58E+04	7.05E+03	2.08E+03	2.47E+03
F ₁₃	1.49E+00	1.09E-01	1.66E+00	1.35E-01	3.94E+00	2.81E-01	1.94E+00	1.46E-01	1.54E+00	2.73E-01
F ₁₄	1.23E+01	3.11E-01	1.30E+01	2.00E-01	1.26E+01	2.83E-01	1.35E+01	2.09E-01	1.23E+01	2.53E-01
F ₁₅	3.51E+02	3.11E-02	3.77E+02	8.02E+01	3.76E+02	7.83E+01	2.12E+02	1.98E+01	4.16E+02	3.74E+01
F ₁₆	1.01E+02	3.11E-03	7.94E+01	2.96E+01	8.57E+01	6.94E+01	1.22E+02	9.19E+01	6.78E+01	2.14E+01
F ₁₇	1.47E+02	3.11E-04	1.37E+02	3.80E+01	7.83E+01	3.76E+01	1.69E+02	1.02E+02	6.51E+01	1.28E+01
F ₁₈	9.04E+02	3.11E-05	9.04E+02	1.08E+01	8.68E+02	6.23E+01	8.20E+02	3.35E+00	9.05E+02	8.59E-01
F ₁₉	9.04E+02	3.11E-06	9.04E+02	1.11E+00	8.74E+02	6.22E+01	8.21E+02	3.35E+00	9.04E+02	3.84E-01
F ₂₀	9.04E+02	3.11E-07	9.04E+02	1.10E+00	8.78E+02	6.03E+01	8.22E+02	4.17E+00	9.04E+02	8.03E-01
F ₂₁	5.00E+02	3.11E-08	5.00E+02	4.80E-13	5.52E+02	1.82E+02	8.33E+02	1.00E+02	5.00E+02	9.36E-06
F ₂₂	8.66E+02	3.11E-09	8.75E+02	1.91E+01	9.36E+02	1.83E+01	5.07E+02	7.26E+00	8.62E+02	2.27E+01
F ₂₃	5.50E+02	3.11E-10	5.34E+02	2.77E-04	5.34E+02	3.57E-03	8.58E+02	6.82E+01	5.34E+02	4.11E-04
F ₂₄	2.00E+02	3.11E-11	2.00E+02	2.85E-14	2.00E+02	6.20E-13	2.13E+02	1.52E+00	2.00E+02	3.33E-14
F ₂₅	2.11E+02	3.11E-12	2.11E+02	7.32E-01	2.14E+02	2.00E+00	2.13E+02	2.55E+00	2.26E+02	8.80E+00

候选池和控制参数值的候选池。在进化过程中,利用在前几代创建试验向量的经验来选择一个策略和参数组合。结果表明 LCoDE 算法可以获得非常优异的优化效果。

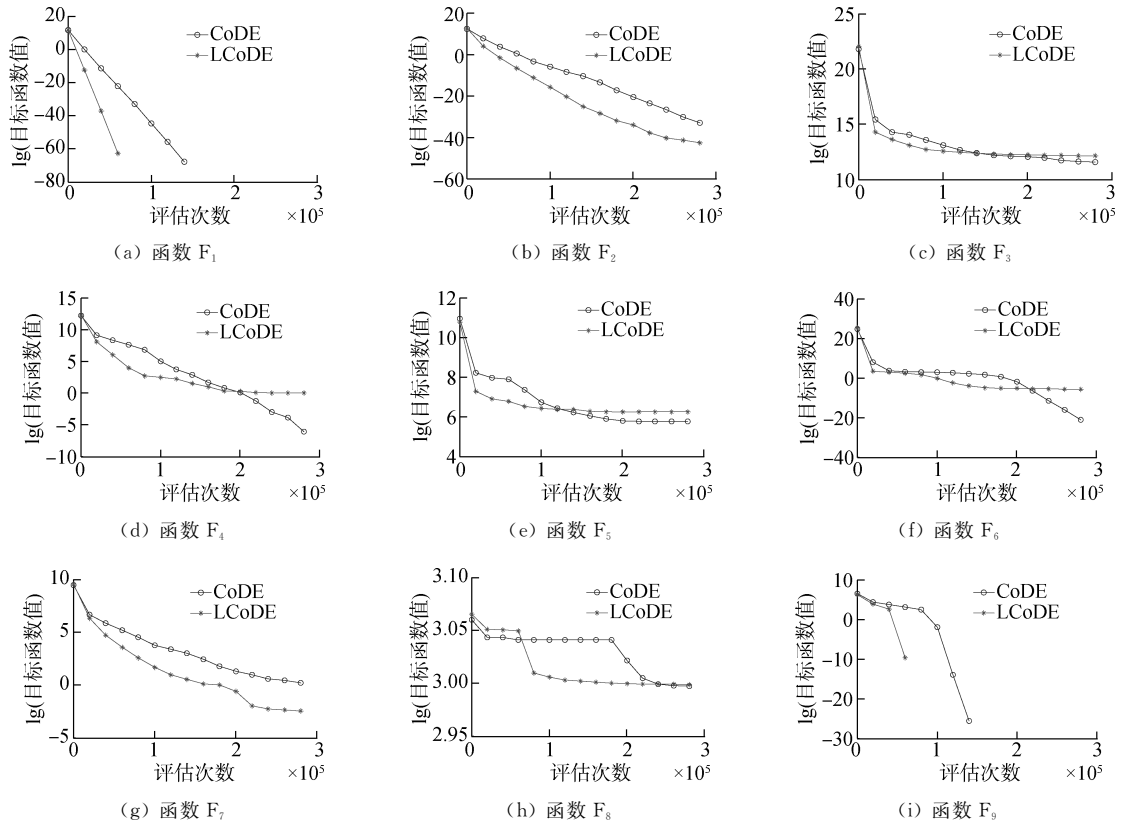


图 1 CoDE 算法和 LCoDE 算法对测试函数 $F_1 \sim F_9$ 的收敛速度对比

3 结 语

针对 CoDE 算法在进化的后期种群会集中在全局最优解附近,过多的个体集聚将浪费计算资源的问题,本文提出使用种群规模线性调节机制来提高算法的收敛性能,在测试函数集上的结果表明该算法具有明显的优势。

参考文献:

- [1] 谢宇. 差分进化的若干问题及其应用研究[D]. 南京:南京理工大学,2015
- [2] Wang Y C. Differential evolution with composite trial vector generation strategies and control parameters[J]. IEEE Transactions on Evolutionary Computation,2011,15(1):55 - 66
- [3] Zhang J Q,Sanderson A C. JADE: Adaptive differential evolution with optional external archive[J]. IEEE Transactions on Evolutionary Computation,2009,13(5):945 - 958
- [4] Brest J,Greiner S,Boskovic B, et al. Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems[J]. IEEE Transactions on Evolutionary Computation,2006,10(6):646 - 657
- [5] Qin A,Suganthan P. Self-adaptive differential evolution algorithm for numerical optimization[C]. 2005 IEEE Congress on Evolutionary Computation,2005:1785 - 1791
- [6] Mallipeddi R,Suganthan P,Pan Q, et al. Differential evolution algorithm with ensemble of parameters and mutation strategies[J]. Applied Soft Computing,2011,11(2):1679 - 1696

(责任编辑:湛 江)