

基于 DE 的浮点数编码自适应进化算法研究^{*}

崔明义

(河南财经学院信息学院 郑州 450002)

摘要 DE 自问世以来,一直是进化计算研究的热点。DE 有着良好的搜索特性、较强的鲁棒性和较高的微调能力。而浮点数编码在提高算法性能和运算质量,将进化算法应用于工程优化领域中有着其它编码所不能比拟的优势。本文将 DE 与浮点数编码的进化计算相结合,提出了基于 DE 的浮点数编码自适应进化算法。经过分析和实验,验证了本算法在搜索性能上是可靠的,方法是可行的,在工程优化中具有较强的实用性。

关键词 DE,浮点数编码,自适应,进化算法

Research on Self-adaptive Float Evolution Algorithm Based on DE

CUI Ming-yi

(School of Information, Henan University of Finance & Economics, Zhengzhou 450002, China)

Abstract Since Differential Evolution (DE) was proposed by researchers, it was the focus in evolution computation research. There are better search performance, stronger robustness and higher perturbation property in DE. But in improving the performance of algorithm and running quality, extending evolution algorithm application to engineering optimization, float code is superior to other codes. In this paper, DE was combined with float code evolution computation, self-adaptive float evolution algorithms based on differential evolution (SFEADE) were proposed. Through analyzing and experiment, the result indicated the algorithm was credible in performance and feasible in method. It was stronger in practicability.

Keywords DE, Float code, Self adaptive, Evolution algorithm

1 引言

1995 年, Rainer Storn 和 Kenneth Price^[1] 提出了将遗传学的分异进化(Differential Evolution, 简称 DE)的启发式方法应用于进化计算。10 余年来,有关 DE 的研究成果不断涌现。1999 年, Rainer Storn^[2] 基于 DE 构建了约束问题的并行优化程序,使 DE 的使用更加方便;2001 年, Abbass 和 Sarker^[3] 等提出了帕累托分异进化(The Pareto Differential Evolution, 简称 PDE)算法,用于多目标优化,解决了两个标准实验问题;2002 年, Abbass^[4] 提出了自适应帕累托分异进化(The Self-adaptive Pareto Differential Evolution, 简称 SPDE)算法,进行自适应遗传操作,提高了多目标优化的竞争性;2003 年, Fan Hui-Yuan 和 J. Lampinen^[5] 将三角变异法作为局部搜索操作用于 DE 算法,改进了算法收敛性和鲁棒性的相互补偿,提高了算法的搜索速度;2004 年, Vesterstrom 和 Thomsen^[6] 用 34 个广泛使用的常见问题将 DE 与粒子群优化(PSO)算法和其他进化算法进行了比较,认为 DE 在优化性能上大大优于这些算法;2005 年, Rainer Storn^[7] 基于 DE 设计了非标准过滤器,使用数字过滤器的最小知识理论解决过滤器设计的最小化问题;同年, Ronkkonen, Kukkonen 和 Price^[8] 在 25 个升级函数的实验床上用 DE 进行实参优化,验证了 DE 的高性能特征。

上述研究者的研究成果极大地丰富了 DE 的性能研究,从实验的角度验证了 DE 良好的搜索性能和应用价值。但由于 DE 的问世才刚刚 10 余年的时间,有关 DE 的理论和应用研究还有较多的令人关注的课题需要探索。将 DE 用于浮点数编码进化算法的研究就是一个十分重要的课题之一,这是

由于以下两个原因:

(1) DE 具有较好的搜索特性、较强的鲁棒性和较高的微调能力,在优化算法的比较中明显地表现出良好的性能^[6]。

(2) 浮点数编码在提高算法性能和运算质量,将进化算法应用于工程优化领域中有着其它编码所不能比拟的优势^[9-11]。

迄今为止,DE 的研究还仅局限于算法本身的性能比较和提高,将 DE 用于浮点数编码自适应进化算法的研究尚未有较大进展。基于上述原因,本文重点研究将 DE 用于浮点数编码自适应进化算法(Self-adaptive Float Evolution Algorithms Based on Differential Evolution, 简称 SFEADE)对算法性能的影响,探索其性能特征,分析其可靠性和可行性,为 DE 的拓展应用开拓新的空间。

2 DE 模式和算法

2.1 DE 模式

DE 作为进化算法的一种新技术,用于连续空间的优化问题。在 DE 中,个体的每个变量值都用实数来表达,这与浮点数编码进化算法相类似。通过某些变量分异规则,保证每一个变量值都在其范围内变化。随机选择三个不同的个体作为双亲个体,其中之一为主双亲。依据一定的概率,主双亲中至少一个变量发生变化,变化量是其他两个双亲的差值乘以一定的比率。如果主双亲变化后优于变化前的主双亲,则新主双亲替换原主双亲,否则原主双亲保留。DE 与传统进化算法(以遗传算法(Genetic Algorithm, 简称 GA)为例)相比有以下不同:

(1) 在遗传算法中,选择两个个体作为双亲,子个体是双

^{*} 本课题受到河南省基础与前沿技术研究项目(082300410100)的资助。崔明义 教授,博士,研究方向为计算智能、软件和网络安全。

亲的交叉组合;而 DE 中,选择三个个体作为双亲,子个体是主双亲的一个摄动(微调)。

(2)在遗传算法中,子个体以某种概率替换双亲,不考虑适应度大小;而 DE 中,只有当子个体优于主双亲时,主双亲才被替换。

设群体规模为 $M(M>4)$,个体由 N 个变量组成,在第 i 代的一个解 l 可表示成多维向量 $X_l^i = (x_{l1}^i, x_{l2}^i, \dots, x_{lN}^i)$,第 k 代群体可表示为集合 $P_k = \{X_1^k, X_2^k, \dots, X_M^k\}$ 。对初始群体, $P_0 = \{X_0^1, X_0^2, \dots, X_0^M\}$,其某个解 l 中变量 j 的分异公式为:

$$x_{j,0}' = \text{lower}(x_j) + \text{rand}_j[0,1] \times (\text{upper}(x_j) - \text{lower}(x_j)) \quad (1)$$

式中, $l=1,2,\dots,M, j=1,2,\dots,N, (\text{lower}(x_j), \text{upper}(x_j))$ 是初始群体个体 l 中变量 j 的取值范围。随机选择四个不同的整数 $r_1, r_2, r_3, r_4 \in [1, M] (r_1 \neq r_2 \neq r_3 \neq r_4)$,以交叉概率 CR 改变子个体每一个变量的值,则

$$\forall j \leq N, x_{j,k}' = \begin{cases} x_{j,k-1}' + F \times (x_{j,r_1,k-1}' - x_{j,r_2,k-1}') & \text{if } (\text{random}[0,1] < CR \wedge j = j_{\text{rand}}) \\ x_{j,k-1}' & \text{otherwise} \end{cases} \quad (2)$$

这里,参数 $F \in (0,1)$ 表示加到主双亲的摄动量, $j_{\text{rand}} \in (1, N)$ 为随机选择变量。如果新个体优于主个体,则被替换,且个体中至少有一个变量被改变。交叉运算后,如果新个体中一个或多个变量超出其范围 $(\text{lower}(x_j), \text{upper}(x_j))$,则进行如下调整:

$$x_{j,k}' = \begin{cases} \frac{x_{j,k}' + \text{lower}(x_j)}{2} & \text{if } x_{j,k}' < \text{lower}(x_j) \\ \text{lower}(x_j) + \frac{x_{j,k}' - \text{upper}(x_j)}{2} & \text{if } x_{j,k}' > \text{upper}(x_j) \\ x_{j,k}' & \text{otherwise} \end{cases} \quad (3)$$

2.2 DE 算法

Step1 输入 $N, M(M>4), F \in (0,1), CR \in [0,1]$, 随机产生第 0 代群体和初始范围 $\text{lower}(x_j), \text{upper}(x_j), j=1,2,\dots,N$;

Step2 对每一个个体 $l \in [1, M]$, 用式(1)求第 0 代群体中个体摄动后的解;

Step3 评价第 0 代群体的个体;

Step4 $k=1$;

Step5 对个体 $l \in [1, M]$, 随机选择四个不同的整数 $r_1, r_2, r_3, r_4 \in [1, M] (r_1 \neq r_2 \neq r_3 \neq r_4)$ 和 $j_{\text{rand}} \in \{1, 2, \dots, N\}$;

Step5.1 对 $j \in [1, N]$, 用式(2)求各变量的修正值;

Step5.2 如果某个或多个变量超出其范围 $(\text{lower}(x_j), \text{upper}(x_j))$, 用式(3)进行调整;

Step5.3 依据下式决定是否替换主双亲:

$$X_k^l = \begin{cases} X' & \text{if } f(X') \leq f(X_{k-1}^l) \\ X_{k-1}^l & \text{otherwise} \end{cases} \quad (4)$$

Step6 评价第 k 代群体的个体;

Step7 判断是否满足终止条件,若未满足, $k=k+1$, 返回 Step5; 否则, 终止, 返回最优解。

3 基于 DE 的浮点数编码自适应进化算法(SFEADE)

显然, DE 中的交叉运算被两个不同个体变量的差分运算所替代, 这与传统进化算法明显不同。此外, DE 模式没有变异运算, 这容易造成算法的“早熟”, 严重影响了算法的优化

质量, 必须依靠个体编码的变异运算抑制“早熟”现象的产生。如果没有变异运算, 算法的收敛结果是很不可靠的。学者 A. E. Eiben 等已经证明^[12]: 当采用浮点数编码时, 有限群体的收敛性唯一地由变异操作所保证。为此, 需对 DE 算法进行改进, 构造基于 DE 的浮点数编码进化算法。构造后的浮点数编码进化算法描述如下:

Step1 输入 $N, M(M>4), F \in (0,1), CR \in [0,1]$, 随机生成第 0 代浮点数编码群体和初始范围 $\text{lower}(x_j), \text{upper}(x_j), j=1,2,\dots,N$;

Step2 对每一个个体 $l \in [1, M]$, 用式(1)求第 0 代群体中个体摄动后的解;

Step3 评价第 0 代群体的个体;

Step4 $k=1$;

Step5 对个体 $l \in [1, M]$, 随机选择四个不同的整数 $r_1, r_2, r_3, r_4 \in [1, M] (r_1 \neq r_2 \neq r_3 \neq r_4)$ 和 $j_{\text{rand}} \in \{1, 2, \dots, N\}$;

Step5.1 对 $j \in [1, N]$, 用式(2)求各变量的修正值;

Step5.2 如果某个或多个变量超出其范围 $(\text{lower}(x_j), \text{upper}(x_j))$, 用式(3)进行调整;

Step5.3 依据下式决定是否替换主双亲:

$$X_k^l = \begin{cases} X' & \text{if } f(X') \leq f(X_{k-1}^l) \\ X_{k-1}^l & \text{otherwise} \end{cases} \quad (5)$$

Step6 依据变异概率 MR 对个体变量进行变异

$$x_{j,k}' = x_{j,k}' + \text{random}(0, 0.01) \times (\text{upper}(x_j) - \text{lower}(x_j)) \quad (6)$$

Step7 评价第 k 代群体的个体;

Step8 判断是否满足终止条件, 若未满足, $k=k+1$, 返回 Step5; 否则, 终止, 返回最优解。

4 应用实例

为了验证 SFEADE 的可靠性和实用性, 本文取两个著名的遗传算法测试函数 F6, F7, 分别对 GA 和 SFEADE 进行了比较实验。这两个函数如下:

(1) F6 函数(Schaffer1 function)

$$\min f(x) = (x_1^2 + x_2^2)^{0.25} [\sin^2(50 \times (x_1^2 + x_2^2)^{0.1}) + 1.0] \quad (7)$$

s. t. $x_i \in [-10, 10]$

(2) F7 函数(Schaffer2 function)

$$\min f(x) = 0.5 + \frac{\sin \sqrt{x_1^2 + x_2^2} - 0.5}{[1 + a \times (x_1^2 + x_2^2)]^2} \quad (8)$$

s. t. $a \in \{0.001, 0.01, 0.1, 1, 1.0\}$

$x_i \in [-10, 10]$

对这两个函数, 均取群体规模为 100, 进化代数为 100。在相同样本下, 分别用 GA 和 SFEADE 测试这两个函数的最佳适应度、平均适应度、标准方差(最优解基因值的偏差指标)等。实验结果如表 1 所示。两个函数的 GA 和 SFEADE 的运行结果分别如图 1、图 2 所示。实验结果表明:

(1) 在相同样本和相同遗传算子下, 同一函数在最佳适应度、平均适应度、标准方差等反映进化性能的指标上, SFEADE 优于 GA, 说明 SFEADE 是可靠和可行的。

(2) SFEADE 的自适应性、鲁棒性、微调和搜索性能, 使其具有比 GA 更强的局部搜索能力, 弥补了 GA 的局部搜索能力差和容易产生“早熟”的不足, 因而可以认为, SFEADE 是一种更优的适合浮点数编码的进化算法。

表1 GA 和 SFEADE 测试结果比较

	F6				F7($\alpha=0.01$)			
	Optimal	Best	Mean	Stddev	Optimal	Best	Mean	Stddev
GA	0	0.018329	0.10887	9.7156e-04	0	0.020627	0.026049	4.1789e-06
SFEADE	0	5.17037e-04	0.001214	7.4643e-04	0	2.35366e-04	84936e-04	3.468e-07

结束语 进化计算一直是计算智能领域研究的热点,但DE的研究相对较晚。10余年来,DE的研究极大地丰富了进化计算的研究成果,为进化计算开辟了新的领域,弥补了传统进化计算的不足。DE以其特有的搜索特性、鲁棒性和微调能力在进化计算中独领风骚。但是,基于DE研究浮点数编

码进化计算的成果还十分罕见。本文利用DE的特性,结合浮点数编码的优势,提出了SFEADE新算法。经过编程和反复实验,表明SFEADE的性能是可靠的,算法是可行的,在工程优化中具有较强的实用性。可以相信,经过众多学者的共同努力,DE的理论和应用研究必将会取得更加丰硕的成果。

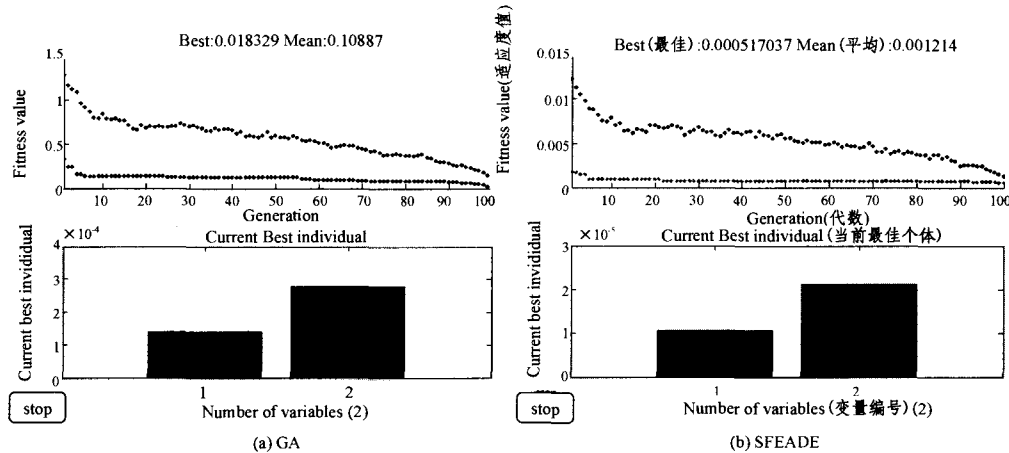


图1 F6函数的GA和SFEADE进化运行结果

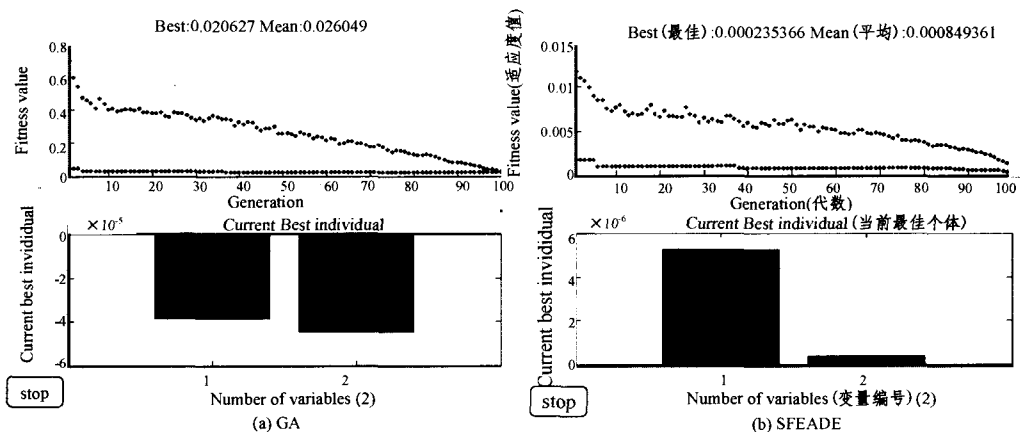


图2 F7函数的GA和SFEADE进化运行结果

参考文献

- [1] Storn R, Price K. Differential evolution - a simple and efficient adaptive scheme for global optimization over continuous spaces. Technical report. International Computer Science Institute, Berkley, 1995
- [2] Storn R. System design by constraint adaptation and differential evolution. Evolutionary Computation. IEEE Transactions on 1999, 3(1): 22-34
- [3] Abbass H A, Sarker R, Newton C. PDE: a Pareto-frontier differential evolution approach for multi-objective optimization problems. Evolutionary Computation // Proceedings of the 2001 Congress on. Volume 2, May 2001; 971-978
- [4] Abbass H A. The self-adaptive Pareto differential evolution algorithm. Evolutionary Computation, CEC'02 // Proceedings of the 2002 Congress on. Volume 1, May 2002; 831-836
- [5] Fan Hui-Yuan, Lampinen J. A Trigonometric Mutation Operation to Differential Evolution. Journal of Global Optimization, 2003, 27(1): 105-129
- [6] Vesterström J, Thomsen R. A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems. Evolutionary Computation, CEC'2004 // Proceedings of the 2004 Congress on. Volume 2, June 2004; 1980-1987
- [7] Storn R. Designing nonstandard filters with differential evolution. Signal Processing Magazine, IEEE, 2005, 22(1): 103-106
- [8] Kukkonen R J, SPrice K V. Real-parameter Optimization with Differential Evolution. Evolutionary Computation // The 2005 IEEE Congress on, Volume 1, Sep. 2005; 506-513
- [9] Michalewicz Z, Krawczyk J, Kazemi M, et al. Genetic Algorithms and Optimal Control Problem // Proc. of 29th IEEE Conf. on Decision and Control, 1990; 1664-1666
- [10] Jomikow C Z, Michalewicz Z. An Experimental Comparison of Binary and Floating Point Representations Genetic Algorithm // Proc. of 4th Int. Conf. on Genetic Algorithms, Morgan Kaufmann, 1991; 31-36
- [11] 雷得明. 多维实数编码遗传算法. 控制与决策, 2000, 15(2): 239-241
- [12] Eiben A E, Arts E H, Van Hee K M. Global Convergence of Genetic Algorithm; an Infinite Markov Chain Analysis // Schwefel H P, Männer R, eds. Parallel Solving from Nature (PPSN 1). Heidelberg Berlin: Springer-Verlag, 1991; 4-12