Linear Regression Strategy for Differential Evolution

José L. Sainz-Pardo

jose.sainz-pardo@umh.es Center of Operations Research, Miguel Hernández University Elche, Spain

ABSTRACT

A new Differential Evolution algorithm is proposed which incorporates a linear regression strategy to find the best single-objective optimization solution. Furthermore, this algorithm contemplates directional information and classical chromosome conservation strategies. Its main motivation is to be used in problems in which it is computationally hard to calculate the fitness differences among solutions.

CCS CONCEPTS

Mathematics of computing;
 Numerical Analysis;

KEYWORDS

global optimization, genetic algorithm, linear regression, evolution strategy

ACM Reference Format:

José L. Sainz-Pardo. 2021. Linear Regression Strategy for Differential Evolution. In 2021 Genetic and Evolutionary Computation Conference Companion (GECCO '21 Companion), July 10–14, 2021, Lille, France. ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3449726.3463287

1 INTRODUCTION

Differential Evolution was introduced for minimizing possibly nonlinear and non-differentiable continuous space functions. Initially, its main advantage was to introduce diversity through differential variation between individuals [3]. Several versions of Differential Evolution algorithms have been designed following diverse strategies, for example [1] has into account directional information too. The greatest contribution of this work is taken into account linear regression information by a Different Evolution algorithm that also considers directional information and other strategies. As far as I know, no article in the literature contemplates this aspect which is specially useful in problems where it is hard to compute the fitness function.

Recently, Differential Evolution and its variants have been applied to several optimization problems related to the COVID-19 pandemic such as [2]. Also related with the COVID-19, a brief computational experience is detailed in Section 5 to calibrate the parameters of *BabSimHospital* simulator. *BabSimHospital* models the number of

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

GECCO '21 Companion, July 10-14, 2021, Lille, France

© 2021 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-8351-6/21/07.

https://doi.org/10.1145/3449726.3463287

ICU beds as function of the number of COVID-19 infected individuals, representing its objective function value the error of the simulation obtained from a set of parameters, the less fitness the better. Since it is hard to compute this fitness, a strategy like linear regression that quickly provides an estimate makes sense.

2 IMPROVING THE POPULATION

The chromosomes of the initial population of size N have been randomly generated within their recommended limits. Once generated each individual i, its fitness is computed and saved in fi[i]. Algorithm 1 describes how to improve a population p. At each crossover three parents are selected and one child is obtained from them. The parents are chosen through fitness proportionate selection (*fps*), but if the number of fitness evaluations is very low then another strategy is to pick the best individual of the population as the first parent. The replacement criterion consists in substituting the worst of the individuals of the population for the child if this is better.

As stopping rule, for example, a *MAX_CALLS* number of fitness evaluations is used.

Algorithm 1: improve_population(p)			
1 nCalls=N			
2 while <i>nCalls</i> < <i>MAX_CALLS</i> do			
$parent_1, parent_2, parent_3 = fps(fi)$			
child=crossover(parent ₁ , parent ₂ , parent ₃)			
4 if <i>fitness</i> (<i>child</i>) < <i>fi</i> [<i>worst individual</i>] then			
5 replace worst individual by child			
6 else			
7 nCalls=nCalls+1			

3 CROSSOVER

In order to use several strategies in the crossover procedure (Algorithm 2), three different amplitude factors F, G and H are used and five different constants are taken into account:

- *CR*₁ determines when the child inherits exactly the same chromosome *i* of one of their parents.
- *CR*₂ determines if the chromosome *i* is obtained as a result of a simple Differential Evolution Formula (Line 8) to increase the diversity of the population.
- *CR*₃ determines if the chromosome *i* is obtained as a result of the formula to incorporate directional information to the population (Line 10).
- *CR*₄ determines if the chromosome *i* is obtained as a result of the formula to incorporate linear regression information to the population (Line 12). Basically, if the coefficient of

	Algorithm 2: crossover	$(parent_1,$	parent ₂ ,	parent ₃)
--	------------------------	--------------	-----------------------	-----------------------

1 F, G, H = random(LOW LIM DE, UP LIM DE)² for i = 1 to numChroms do $(\hat{x}, R^2) = min \ LR(evaluations, i)$ 3 r = random(0, 1)4 if $r < CR_1$ then 5 $child[i] = fps(parent_1[i], parent_2[i], parent_3[i])$ 6 else if $r < CR_2$ then 7 $child[i] = parent_1[i] + F \cdot (parent_2[i] - parent_1[i])$ 8 else if $r < CR_3$ then 9 $child[i] = parent_1[i] + G \cdot (parent_2[i] - parent_3[i])$ 10 else if $r < CR_4$ and $R^2 > MIN_R^2$ then 11 $child[i] = parent_1[i] + H \cdot (\hat{x} - parent_1[i])$ 12 else 13 child[i] = random(lower[i], upper[i]) 14 if $random(0, 1) < CR_5$ then 15 d = random(1, MAX DEC)16 $micro_step = 10^{-d}random(-1, 1)$ 17 $child[i] = child[i] + micro_step$ 18 **if** child[i] < lower[i] **or** child[i] > upper[i] **then** 19 child[i] = random(lower[i], upper[i])20

determination R^2 is relevant, directional vectors are incorporated towards the solution found \hat{x} which minimizes the fitness estimate. *min_LR(evaluations)* procedure is explained in Section 4.

• *CR*₅ decides if small differences in the chromosomes are introduced.

As it can be noted, $0 \le CR_1 < CR_2 < CR_3 < CR_4 \le 1$. Finally, since some chromosomes could be outside the recommended limits, these are generated randomly and correctly.

4 LINEAR REGRESSION STRATEGY

In optimization problems where it is hard to calculate the fitness differences between two solutions, it could be effective to incorporate directional information towards a solution which minimizes a fitness estimate. A common technique to find estimates between a quantitative response and one or more explanatory variables is linear regression. Algorithm 3 describes how it is obtained the value for the chromosome of interest vObj from all the evaluations already performed by a linear regression technique. The different chromosomes are now the variables of the linear regression. First of all it is shaped the set of candidate variables, they can be considered as candidates: single variables, interactions between variables, quadratic terms and others. From the candidate variables are selected MAX_VARS variables x by a stepwise method like forward selection. In Line 3 it is obtained the linear regression function f(x) which is minimized in the following line. For minimizing f(x) the variables must be bounded among their recommended limits and any method of continuous minimization can be used like

Differential Evolution too. Finally, the value corresponding to the chromosome of interest vObj and the determination coefficient R^2 are returned. Note that have not been obtained estimates for all the variables, but the forced inclusion of vObj guaranteed its estimate.

Algorithm 3: linear_regression (evaluations, vObj)
1 <i>candidates</i> = { single variables \cup interactions \cup quadratic
terms } 2 $x = \{vOb \ i \cup forward \ selection(candidates, MAX \ VARS)\}$
$R^{2}, f) = LR(evaluations, x)$
$4 \ \hat{x} = \min \ f(x)$
5 return $\hat{x}[vObj]$

5 COMPUTATIONAL EXPERIENCE

Two computational experiments have been developed to find the best parameters for the *BabSimHospital* simulator. The constants have been set by: N = 25, $LOW_LIM_DE = 0.2$, $UP_LIM_DE = 1.8$, $CR_1 = 0.2$, $CR_2 = 0.4$, $CR_3 = 0.6$, $CR_5 = 0.1$ and $MAX_ITER = 200$. In order to measure the effect of the linear regression strategy CR_4 was set at two different values $CR_4 = 0.6$ (without linear regression strategy) and $CR_4 = 0.8$. Regarding the linear regression, $MIN_R2 = 0.3$, single variables and only interactions with the variable of interest were considered candidates and just three variables were selected ($MAX_VARS = 3$). Table 1 summarizes the obtained results for both experiments being *best fitness* and *worst fitness* the results respectively obtained by the best and the worst solution of the final population.

CR_4	best fitness	worst fitness
0.6	16.91309	28.33395
0.8	15.91872	26.05241

Table 1: Computational Experience Summary

6 DISCUSSION AND FUTURE WORKS

A new Differential Evolution algorithm has been proposed that incorporates directional information towards chromosomes estimated by linear regression. The computational experience performed could indicate advantage of the introduced method both to improve the best solution and to improve the rest of individuals of the population. In future works, more extensive computational experiments will be carried out on diverse instances in order to check the validity and scope of this proposal. Furthermore, in some areas it can be studied the effect of already contrasted regression models instead of obtaining these on the fly.

REFERENCES

- A.W. Iorio and X. Li. 2006. Incorporating directional information within a Differential Evolution algorithm for multi-objective optimization. *Proceeding of the Genetic and Evolutionary Computation Conference 2006* (2006), 691–697.
 J.L. Sainz-Pardo and J. Valero. 2020. COVID-19 and other viruses: holding back its
- expansion by massive testing. *arXiv* (2020). https://arxiv.org/abs/2012.12345
- [3] R. Storn. 1996. On the usage of differential evolution for function optimization. Proceedings of the 1996 Biennial Conference of the North American Fuzzy Information Processing Society (1996), 519–523.