

# Uporaba regresijskih metod v diferencialni evoluciji

Iztok Fister Jr., Janez Brest

Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko, Smetanova 17, 2000 Maribor  
E-pošta: iztok.fister2@uni-mb.si

## Abstract

Nowadays, differential evolution is one of the most powerful methods for optimization that supports several strategies for generating new trial solution. In order to predict the best strategy during the run, this paper proposes use of various regression methods on ensemble strategies in differential evolution. The preliminary results of this algorithm on a well-known test suite were shown that this hybridization improves the results of the original differential evolution significantly.

## 1 Uvod

Diferencialna evolucija (DE) je dandanes ena izmed popularnih in najučinkovitejših optimizacijskih metod. Ta algoritem sta odkrila Storn in Price [10]. DE spada med populacijske algoritme, kjer so posamezniki predstavljeni z realnimi vrednostmi. Vsak posameznik predstavlja mogočo rešitev problema, ki ga rešujemo. Ta rešitev je podvržena operatorjema mutacije in križanja. S pomočjo operatorja selekcije iz populacije izberemo najboljšega posameznika glede na njegovo vrednost uspešnosti.

Skozi čas so se pa pojavile različne modifikacije klasičnega algoritma diferencialne evolucije. Brest in ostali so, na primer, leta 2006 pokazali samoprilagodljivo diferencialno evolucijo jDE [2], kjer se parametri spreminjajo med samim tekom algoritma. Posebej zanimive so danes modifikacije algoritma, ki operirajo z ansambli parametrov in strategij [8]. Celoten pregled dela na tem področju je zbran v delu [3].

Odločanje na osnovi podatkov iz več virov je že dolgotrajna praksa v vsakdanjem človekovem življenju [9]. Z odločanjem se danes ukvarjata predvsem področji računske inteligence (angl. Computer Intelligence, krajše CI) in strojnega učenja (angl. Machine Learning, krajše ML). Posebej popularno je danes postalo t.i. ansambelsko učenje (angl. Ensemble Learning), ki ga lahko uporabimo tudi za napovedovanje najprimernejših regresijskih vrednosti dobljenih od ansambla strategij DE. Začetne raziskave na tem področju in preliminarne rezultate smo predstavili v [5]. Namesto originalnega potomca dobljenega z delovanjem določene strategije DE, njegovo vlogo v evolucijskem procesu prevzema predviden regresijski vektor.

Struktura članka je naslednja: v naslednjem poglavju prikažemo ozadje potrebno za razumevanje predlaganega

algoritma. V poglavju 3 povemo nekaj o regresijskih metodah, ki jih uporabljamo za hibridizacijo z diferencialno evolucijo. Poglavje 4 prikazuje eksperimente z rezultati. Članek zaključimo s povzetkom opravljenega dela in načrtamo smeri za nadaljnje delo.

## 2 Diferencialna evolucija

Diferencialna evolucija je evolucijski algoritem primeren za zvezno in kombinatorično optimizacijo. To je populacijski algoritem, ki sestoji iz  $NP$  vektorjev realnih števil, t.j. mogočih rešitev:

$$\mathbf{x}_i^{(t)} = (x_{i1}^{(t)}, \dots, x_{iD}^{(t)}), \quad \text{for } i = 1 \dots NP, \quad (1)$$

kjer  $D$  označuje dimenzije problema,  $t$  pa generacijo.

DE uporablja operatorje mutacijo, križanje in selekcijo. Pri mutaciji naključno izberemo dve rešitvi in prištejemo skalirano razliko med njima k tretji rešitvi. Mutacijo matematično predstavimo kot:

$$\mathbf{u}_i^{(t)} = \mathbf{x}_{r0}^{(t)} + F \cdot (\mathbf{x}_{r1}^{(t)} - \mathbf{x}_{r2}^{(t)}), \quad \text{for } i = 1 \dots NP, \quad (2)$$

kjer  $F \in [0.1, 1.0]$  označuje skalirni faktor, ki določa velikost modifikacije, medtem ko so  $r0$ ,  $r1$ ,  $r2$  naključno izbrane celoštevilске vrednosti v intervalu  $1 \dots NP$ .

Križanje v diferencialni evoluciji matematično predstavimo kot:

$$w_{i,j} = \begin{cases} u_{i,j}^{(t)} & \text{rand}_j(0, 1) \leq CR \vee j = j_{rand}, \\ x_{i,j}^{(t)} & \text{otherwise,} \end{cases} \quad (3)$$

kjer  $CR \in [0.0, 1.0]$  določa verjetnost kopiranja  $j$ -tega parametra k testni rešitvi. Omenimo, da relacija  $j = j_{rand}$  zagotavlja, da je testni vektor različen od originalne rešitve  $\mathbf{x}_i^{(t)}$ .

Matematično selekcijo v diferencialni evoluciji predstavimo kot:

$$\mathbf{x}_i^{(t+1)} = \begin{cases} \mathbf{w}_i^{(t)} & \text{if } f(\mathbf{w}_i^{(t)}) \leq f(\mathbf{x}_i^{(t)}), \\ \mathbf{x}_i^{(t)} & \text{otherwise.} \end{cases} \quad (4)$$

Križanje in mutacijo lahko v tehničnem smislu izvedemo v diferencialni evoluciji na več načinov, ki jih imenujemo tudi strategije in označimo s posebno notacijo (npr. 'rand/1/bin'). Zbirko različnih strategij DE imenujemo tudi ansambel strategij (angl. Ensemble Strategies, krajše ES).

### 3 Regresije

V naših testih uporabljamo naslednje regresijske metode:

- random forest (krajše RF) [1, 4],
- ekstremna naključna drevesa (angl. Extremely Randomized Trees, krajše EXT) [7],
- 'gradient boosting' (krajše GB) [6],
- odločitvena drevesa (angl. Decision Trees, krajše DT) in
- splošni linearni model (angl. Generalized Linear Model, krajše GLM) [11].

### 4 Predlagani algoritem RDE

Bistveni del našega dela je združitev regresijskih metod, ki spadajo med metode strojnega učenja, in diferencialne evolucije, ki spada med računsko inteligenco.

---

**Algorithm 1** Algoritem RDE

---

```
1: Initialize the DE population  $\mathbf{x}_i = (x_{i1}, \dots, x_{iD})$ 
2: repeat
3:   for  $i = 1$  to  $i \leq NP$ 
4:     Create test set  $T$  on vector  $\mathbf{x}_i$  using ES (Apply Alg. 2);
5:     Create validation set  $\mathbf{v}_i$  applying 'rand/1/bin' on  $\mathbf{x}_i$ ;
6:     Build regression vector  $\mathbf{r}_i$  applying the regression method
       using  $T$  and  $\mathbf{v}_i$ ;
7:     if ( $f(\mathbf{r}_i) < f(\mathbf{x}_i)$ ) Insert  $\mathbf{r}_i$  into  $Q$ ;
8:     else Insert  $\mathbf{x}_i$  into  $Q$ ;
9:     end if
10:  endfor
11:   $P = Q$ ;
12: until (Termination condition meet)
```

---

Funkcija za ustvarjanje testne množice začne s prazno množico. Za vsako od strategij v ansamblu  $s \in ES$  ustvarimo vektor  $\mathbf{t}$ , ki ga dodamo testni množici  $T$ . ES vključuje 10 osnovnih strategij algoritma DE [10, 3].

---

**Algorithm 2** Ustvarjanje testne množice

---

```
1:  $T = \emptyset$ ;
2: forall  $s \in ES$ 
3:   Create vector  $\mathbf{t}$  using ensemble strategy  $s$  on vector  $\mathbf{x}_i$ ;
4:   Add vector  $\mathbf{t}$  to test set  $T$ ;
5: endfor
```

---

### 5 Eksperimenti in rezultati

Cilj našega eksperimentalnega dela je bil pokazati, kako na rezultate optimizacije funkcij vplivajo različne regresijske metode. V ta namen smo originalni algoritem DE hibridizirali z regresijskimi metodami RF, EXT, GB, DT in GLM in uporabili pri optimizaciji standardnih funkcij dimenzije  $D = 10$  (tabela 1).

Rezultati preliminarjih testov so pokazali, da uporaba regresijskih metod na ES v DE izboljšuje rezultate originalnega DE. Še več, regresijska metoda RF rezultate izboljša celo signifikantno.

Tabela 1: Testna množica funkcij

$f$	Funkcija	Območje
$f_1$	Rosenbrock	$-15.00 \leq x_i \leq 15.00$
$f_2$	Rastrigin	$-15.00 \leq x_i \leq 15.00$
$f_3$	Sphere	$-100.00 \leq x_i \leq 100.00$
$f_4$	Griewangk	$-600 \leq x_i \leq 600$
$f_5$	Ackley	$-32.00 \leq x_i \leq 32.00$

### 6 Zaključek

V tem članku je bil na kratko opisan algoritem, ki združuje algoritem diferencialne evolucije z regresijskimi metodami. Eksperimenti so pokazali, da regresijske metode dajo velik doprinos k zmogljivosti klasične diferencialne evolucije. V prihodnosti želimo dograditi regresijske metode tudi ostalim modifikacijam diferencialne evolucije, npr. jDE.

### Literatura

- [1] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.
- [2] J. Brest, S. Greiner, B. Bošković, M. Mernik, and V. Žumer. Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems. *Evolutionary Computation, IEEE Transactions on*, 10(6):646–657, 2006.
- [3] S. Das and P.N. Suganthan. Differential evolution: A survey of the state-of-the-art. *Evolutionary Computation, IEEE Transactions on*, 15(1):4–31, 2011.
- [4] I. Jr. Fister, D. Fister, and I. Fister. Differential evolution strategies with random forest regression in the bat algorithm. In *Proceedings of the fifteenth international conference on Genetic and evolutionary computation conference companion*. ACM, 2013 in press.
- [5] I. Jr. Fister, I. Fister, and J. Brest. Comparing various regression methods on ensemble strategies in differential evolution. In *Proceedings of 19th International Conference on Soft Computing MENDEL*, pages 87–92, 2013.
- [6] J.H. Friedman. Stochastic gradient boosting. *Computational Statistics & Data Analysis*, 38(4):367–378, 2002.
- [7] P. Geurts, D. Ernst, and L. Wehenkel. Extremely randomized trees. *Machine learning*, 63(1):3–42, 2006.
- [8] R. Mallipeddi, P.N. Suganthan, Q.K. Pan, and M.F. Tasgetiren. Differential evolution algorithm with ensemble of parameters and mutation strategies. *Applied Soft Computing*, 11(2):1679 – 1696, 2011.
- [9] R. Polikar. Ensemble learning. In C. Zhang and Y. Ma, editors, *Ensemble Machine Learning*, pages 1–34. Springer-Verlag, Berlin, 2012.
- [10] R. Storn and K. Price. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, 11(4):341–359, 1997.
- [11] A.N. Tikhonov, A.S. Leonov, and A.G. Yagola. *Nonlinear Ill-Posed Problems*. Chapman and Hall, 1998.