

Študija samoprilagajanja krmilnih parametrov pri algoritmu DEMOwSA

Aleš Zamuda, Janez Brest, Borko Bošković, Viljem Žumer

Univerza v Mariboru

Fakulteta za elektrotehniko, računalništvo in informatiko

Inštitut za računalništvo

Smetanova 17, 2000 Maribor, Slovenija

E-pošta: ales.zamuda@uni-mb.si, janez.brest@uni-mb.si, borko.boskovic@uni-mb.si, zumer@uni-mb.si

Povzetek. V članku predstavljamo študijo samoprilagodljivih krmilnih parametrov algoritma diferencialne evolucije za večkriterijsko optimizacijo, ki ga krmili samoprilagoditveni mehanizem, predstavljen v evolucijskih strategijah. Samoprilagajanje parametrov omogoča danemu evolucijskemu algoritmu učinkovitejše iskanje, saj se algoritem lahko prilagodi optimizacijskemu problemu, ki ga rešuje. Z eksperimentom prikažemo dejanske vrednosti in spreminjanje samoprilagodljivih krmilnih parametrov na znanih testnih funkcijah.

Glavne besede: evolucijsko računanje, diferencialna evolucija, večkriterijska optimizacija, samoprilagoditev

A Study in Self-adaptation of Control Parameters in the DEMOwSA Algorithm

Extended Abstract. In this paper we present an experimental analysis showing that the self-adaptation of control parameters plays an important role in the multiobjective optimization process (refer to Figure 1 for notion of multiobjective optimality). Experimental results of a self-adaptive differential evolution algorithm are evaluated on the set of benchmark functions provided for the CEC 2007 Special session on Performance Assessment & Competition on Multi-objective Optimization Algorithms, as seen in Tables 2–7. Self-adaptation is proven to statistically outperform fixed parameters, using t-test on the empirical results in these tables. The values of control parameters are encoded in each individual (see Figure 2) and changed during the optimization process. They depend on the nature of the problem being solved, as can be seen in Table 1 and Figures 3 and 4 which show how using self-adaptation good control parameters are obtained to improve the search results.

Key words: evolutionary computation, differential evolution, multi-objective optimization, self-adaptation

1 Uvod

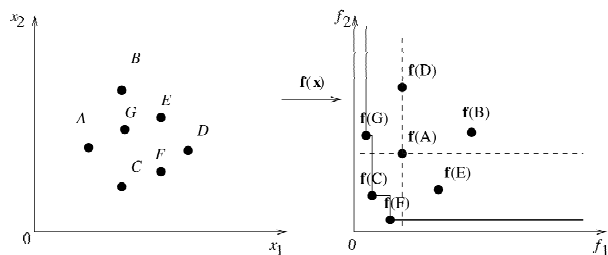
V članku predstavljamo študijo samoprilagajanja krmilnih parametrov, ki smo jo izvedli na testnih funkcijah in s pomočjo orodij, ki so bila predlagana na posebni sekciji za oceno učinkovitosti večkriterijskih algoritmov za optimizacijo [13]. Za študijo smo uporabili naš algoritem, ki je bil prvič predstavljen na mednarodni konferenci evolucijskega računanja CEC 2007 [21].

Prispevek je organiziran takole. V drugem poglavju predstavimo večkriterijsko optimizacijo in pristope za reševanje večkriterijskih problemov. Tretje poglavje govori o diferencialni evoluciji za večkriterijsko optimizacijo s samoprilagoditvenim mehanizmom za krmilne parametre. V četrtem poglavju opisujemo nastavitve parametrov in eksperimentalne rezultate pri večkriterijski optimizaciji na izbranih testnih funkcijah. Sledi še sklopno poglavje z idejami za nadaljnje delo.

2 Večkriterijska optimizacija

V vsakdanjem življenju se pri odločanju pogosto srečujemo z zahtevo po izbiri različnih možnosti. To lažje opravimo s postavitvijo kriterijev. Ker so si ti kriteriji pogosto v nasprotju, izboljšanje ocene rešitve po enem kriteriju lahko povzroči poslabšanje ocene po drugih kriterijih. Tedaj ena sama rešitev ni optimalna po vseh kriterijih, temveč se srečamo z množico optimalnih rešitev. Za lažjo izbiro ene same od rešitev problema pogosto želimo dobro spoznati, kakšne možnosti sploh imamo, oz. kakšne najboljše možnosti imamo, ter se šele pozneje odločiti za eno od njih. Ta pristop opisuje večkriterijsko optimizacijo, tj. sočasno optimizacijo več konfliktnih kriterijev.

Problem večkriterijske optimizacije (ang. *multiobjective optimization problem* – MOP) definiramo kot iskanje dopustnega vektorja spremenljivk oz. parametrov $\mathbf{x} = (x_1, x_2, \dots, x_D)$, ki optimizira (minimizira) vek-



Slika 1. Dominantnost na primeru večkriterijske funkcije $f(\mathbf{x})$
Figure 1. Dominance of multiobjective function $f(\mathbf{x})$

torsko funkcijo: $f(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_M(\mathbf{x}))$ in spremenljivke zadoščajo m neenakostnim omejitvam: $g_i(\mathbf{x}) \geq 0$; $i = 1, 2, \dots, m$.

Rešitev večkriterijskega optimizacijskega problema \mathbf{x} dominira rešitev \mathbf{y} ($\mathbf{x} \preceq \mathbf{y}$), če velja, da rešitev \mathbf{x} ni slabša od rešitve \mathbf{y} po vseh kriterijih ($f_i(\mathbf{x}) \leq f_i(\mathbf{y})$, $\forall i = 1, \dots, M$) in je \mathbf{x} boljša od rešitve \mathbf{y} po vsaj enem kriteriju ($\exists j \in \{1, \dots, M\}: f_j(\mathbf{x}) < f_j(\mathbf{y})$). Prostor kriterijev \mathbb{R}^M (M je št. kriterijev) je z relacijo \preceq delno urejen, saj nekatere rešitve medsebojno niso primerljive. Za slikovno razjasnitev pojma dominantnosti glej sliko 1. Kot je videti iz slike, je rešitev A dominantna (boljša) od B in D , neprimerljiva z E in G , ter slabša od C in F .

Množica nedominiranih rešitev, imenovana tudi *nedominirana fronta*, iz množice rešitev P je podmnožica vseh tistih rešitev iz P , ki jih nobena rešitev iz množice P ne dominira. Podmnožica nedominiranih rešitev celotnega prostora dopustnih rešitev se imenuje *optimalna fronta Pareto**, njeni elementi pa *optimalne rešitve Pareto*. Množico nedominiranih rešitev na sliki 1 tvorijo rešitve G , C in F , saj niso dominirane z nobeno drugo rešitvijo, tj. $f(G)$, $f(C)$ in $f(F)$ so manjše od drugih po vsaj enem kriteriju. Osnovna cilja večkriterijske optimizacije sta porazdelitev in bližina takih rešitev glede na fronto Pareto.

V začetku večkriterijske optimizacije so se uporabljale klasične enokriterijske optimizacijske metode, pri katerih se je večkriterijski problem z utežno funkcijo preslikal v enokriterijski problem [17]. Pozneje se je pri večkriterijski optimizaciji precej povečal interes za

*Princip optimalnosti, imenovan po Vilfredu Pareto, 1848–1923.

$x_{1,1,G}$...	$x_{1,D,G}$	$F_{1,G}$	$CR_{1,G}$	$f_1^I(x_{1,G})$...	$f_M^I(x_{1,G})$
$x_{2,1,G}$...	$x_{2,D,G}$	$F_{2,G}$	$CR_{2,G}$	$f_1^I(x_{2,G})$...	$f_M^I(x_{2,G})$
$x_{3,1,G}$...	$x_{3,D,G}$	$F_{3,G}$	$CR_{3,G}$	$f_1^I(x_{3,G})$...	$f_M^I(x_{3,G})$
\vdots	\ddots	\vdots	\vdots	\vdots	\vdots	\ddots	\vdots
$x_{NP,1,G}$...	$x_{NP,D,G}$	$F_{NP,G}$	$CR_{NP,G}$	$f_1^I(x_{NP,G})$...	$f_M^I(x_{NP,G})$

Slika 2. Predstavitev samoprilagodljivih krmilnih parametrov
Figure 2. Self-adaptive control parameters presentation

Tabela 1. Vrednosti krmilnih parametrov po $5e+5$ FES
Table 1. Control parameters values after $5e+5$ FES

Zap. št.	Funkcija	\bar{F}_{imp}	\bar{CR}_{imp}
1.	OKA2	0,53 (0,32)	0,40 (0,19)
2.	SYMPART	0,17 (0,06)	0,75 (0,16)
3.	S_ZDT1	0,26 (0,08)	0,55 (0,14)
4.	S_ZDT2	0,31 (0,09)	0,56 (0,14)
5.	S_ZDT4	0,32 (0,10)	0,55 (0,14)
6.	R_ZDT4	0,45 (0,23)	0,26 (0,17)
7.	S_ZDT6	0,55 (0,00)	0,40 (0,00)
8.	S_DTLZ2_M3	0,32 (0,09)	0,49 (0,13)
9.	R_DTLZ2_M3	0,68 (0,15)	0,35 (0,11)
10.	S_DTLZ3_M3	0,33 (0,10)	0,48 (0,12)
11.	WFG1_M3	0,52 (0,16)	0,24 (0,08)
12.	WFG8_M3	0,35 (0,12)	0,46 (0,13)
13.	WFG9_M3	0,63 (0,18)	0,31 (0,10)
14.	S_DTLZ2_M5	0,36 (0,11)	0,32 (0,09)
15.	R_DTLZ2_M5	0,61 (0,16)	0,23 (0,07)
16.	S_DTLZ3_M5	0,62 (0,16)	0,06 (0,02)
17.	WFG1_M5	0,63 (0,16)	0,24 (0,07)
18.	WFG8_M5	0,54 (0,15)	0,40 (0,10)
19.	WFG9_M5	0,66 (0,16)	0,27 (0,08)

uporabo populacijskih iskalnih algoritmov [8, 9, 10, 22, 1, 23, 16], kot so evoliucijski algoritmi, ki so se pokazali kot zelo uporabni pri reševanju kompleksnih večkriterijskih optimizacijskih problemov [9].

3 Diferencialna evolucija za večkriterijsko optimizacijo s samoprilagajanjem

Za namen te študije smo uporabili algoritem za večkriterijsko optimizacijo DEMOWSA [21, 20]. Izvorni kod in idejna zasnova algoritma temeljita na algoritmu DEMO [16, 19], ki vsebuje večkriterijske selekcijske strategije NSGA-II [10], SPEA2 [22] in IBEA [23]. Poleg selekcijskih strategij algoritem DEMO sestavlja še osnovni algoritem diferencialne evolucije (DE) [18]. Ker gre za osnovno varianto DE, DEMO nima mehanizma za samoprilagajanje krmilnih parametrov v DE. DE s samoprilagajanjem krmilnih parametrov deluje bolje in robustneje [15, 6, 5, 7]. Pristop s samoprilagajanjem z varianto algoritma diferencialne evolucije so izbrali tudi drugi algoritmi za večkriterijsko optimizacijo [1, 2].

V našem algoritmu DEMOWSA smo implementirali nov način mehanizma samoprilagajanja, ki je logaritmično normalno porazdeljeno samoprilagajanje krmilnih parametrov, znano iz evoliucijskih strategij (ES) [4, 3]. V ES so krmilni parametri vključeni v posameznike skupaj z iskalnimi parametri. Vsak posameznik (glej sliko 2) algoritma DEMOWSA je razširjen s samoprilagodljivimi krmilnimi parametri F in CR . Vrednosti teh parametrov se prilagajajo med evoliucijskim procesom. V tem procesu izdelamo nov primerek iz starševskih primerkov v populaciji s pomočjo operatorjev selekcije,

Tabela 2. Rezultati za indikator R na testnih funkcijah 1–7
Table 2. Results for R indicator on the benchmark functions 1–7

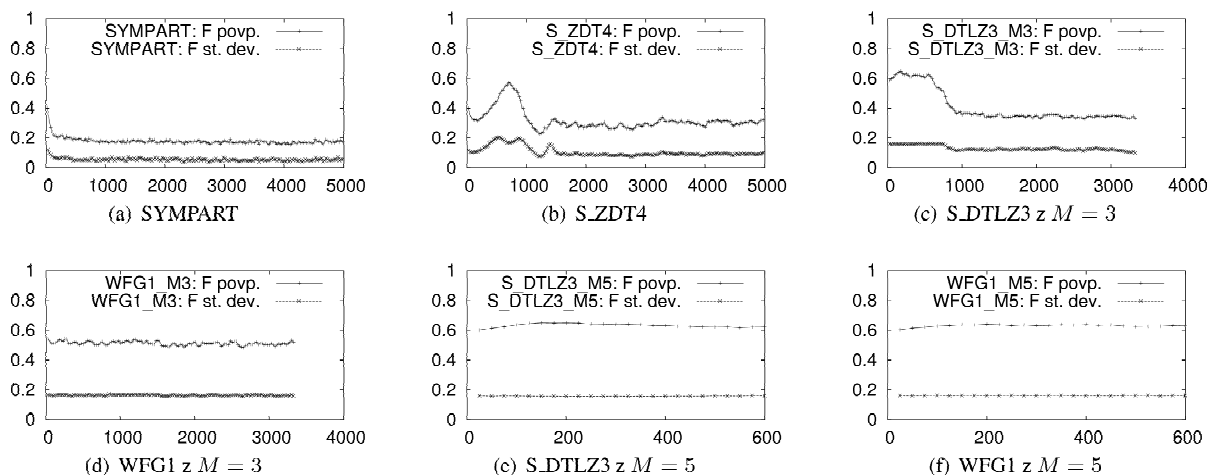
	1. OKA2	2. SYMPART	3. S_ZDT1	4. S_ZDT2	5. S_ZDT4	6. R_ZDT4	7. S_ZDT6
Najboljši	-9,9912e-04	1,2418e-06	3,5832e-09	8,8060e-08	-5,1506e-09	1,1625e-04	-1,0216e-06
Srednji	-9,1005e-04	1,5838e-06	4,9212e-07	1,2702e-06	3,0912e-04	5,2437e-04	-1,0216e-06
Najslabši	-6,5077e-05	2,0972e-06	5,4084e-06	4,0190e-02	6,8877e-04	1,4756e-03	4,5350e-02
Povprečje	-8,4180e-04 [‡]	1,5972e-06[†]	9,3739e-07[†]	1,7634e-02	2,6829e-04[†]	5,7458e-04 [‡]	3,6224e-03 [‡]
Standardni odklon	2,2104e-04	1,9206e-07	1,2040e-06	2,0304e-02	1,8110e-04	3,1949e-04	1,2536e-02

Tabela 3. Rezultati za indikator R na testnih funkcijah 8–13 z $M = 3$
Table 3. Results for R indicator on the benchmark functions 8–13 when $M = 3$

	8. S_DTLZ2	9. R_DTLZ2	10. S_DTLZ3	11. WFG1	12. WFG8	13. WFG9
Najboljši	3,1669e-06	1,9488e-04	1,7962e-07	3,7746e-02	-2,6671e-02	-8,9602e-03
Srednji	1,9806e-05	2,7225e-04	1,3207e-06	4,1318e-02	-2,5825e-02	-7,8284e-03
Najslabši	3,7352e-05	1,8596e-03	5,9104e-06	4,3197e-02	-2,4263e-02	-6,3708e-03
Povprečje	1,9844e-05[†]	4,4428e-04[†]	1,6725e-06[†]	4,0943e-02[†]	-2,5818e-02 [‡]	-7,8520e-03[†]
Standardni odklon	8,9721e-06	3,9119e-04	1,4308e-06	1,6153e-03	6,0301e-04	6,4689e-04

Tabela 4. Rezultati za indikator R na testnih funkcijah 8–13 z $M = 5$
Table 4. Results for R indicator on the benchmark functions 8–13 when $M = 5$

	8. S_DTLZ2	9. R_DTLZ2	10. S_DTLZ3	11. WFG1	12. WFG8	13. WFG9
Najboljši	7,0462e-06	7,3698e-05	1,3937e-06	4,5506e-02	-1,3866e-03	1,3504e-03
Srednji	1,0496e-05	9,3354e-05	1,9857e-06	4,6004e-02	9,1131e-04	2,4433e-03
Najslabši	1,8683e-05	2,2076e-04	3,0883e-06	4,6334e-02	2,7408e-03	3,2385e-03
Povprečje	1,0820e-05 [‡]	9,8614e-05[†]	2,0119e-06[†]	4,5998e-02 [‡]	6,5218e-04 [‡]	2,3653e-03[†]
Standardni odklon	2,7352e-06	3,0240e-05	4,3690e-07	2,2415e-04	1,0519e-03	4,9068e-04



Slika 3. Na 25 generacij povprečna vrednost povprečne izboljšane vrednosti krmilnega parametra F za 25 neodvisnih zagonov
Figure 3. Each 25 generations-averaged value of the average improved value for control parameter F of 25 independent runs

križanja in mutacije.

Glavna razlika med algoritmoma DEMOwSA in DEMO je v operatorju mutacije. DEMOwSA za ustvarjanje novega posameznika za generacijo $G + 1$ uporablja samoprilagodljivi faktor ojačenja diferenčnega vektorja F_i za vsakega od posameznikov i iz generacije G :

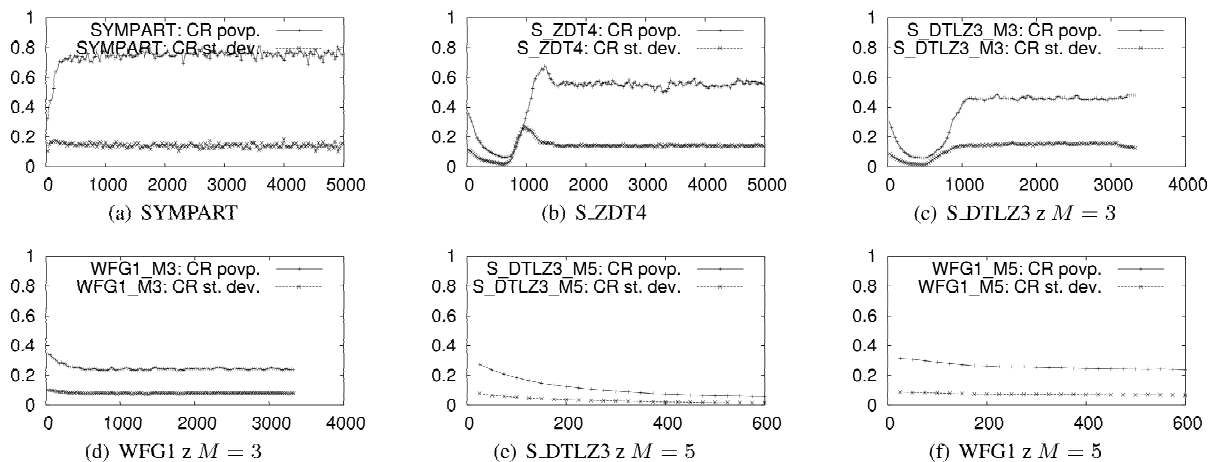
$$F_{i,G+1} = \langle F_G \rangle_i \times e^{\tau N(0,1)},$$

Tu τ označuje stopnjo učenja in je ponavadi sorazmeren

s $\tau \sim 1/\sqrt{2D}$ ter D označuje dimenzijo problema. $N(0,1)$ je po Gaussu normalno porazdeljeno naključno število. Izraz $\langle F_G \rangle_i$ označuje povprečenje parametra F i -tega obravnavanega posameznika:

$$\langle F_G \rangle_i = (F_{i,G} + F_{r_1,G} + F_{r_2,G} + F_{r_3,G})/4.$$

Vrednosti r_1 , r_2 in r_3 označujejo indekse uniformno naključno izbranih posameznikov iz generacije G , ki sodelujejo pri ustvarjanju novega posameznika v mutaci-



Slika 4. Na 25 generacij povprečna vrednost povprečne izboljšane vrednosti krmilnega parametra CR za 25 neodvisnih zagonov
Figure 4. Each 25 generations-averaged value of the average improved value for the control parameter CR of 25 independent runs

Tabela 5. Rezultati za indikator $I_{\overline{H}}$ na testnih funkcijah 1–7
Table 5. Results for $I_{\overline{H}}$ indicator on the benchmark functions 1–7

	1. OKA2	2. SYMPART	3. S.ZDT1	4. S.ZDT2	5. S.ZDT4	6. R.ZDT4	7. S.ZDT6
Najboljši	5,2590e-03	3,6018e-06	1,3953e-04	1,9139e-04	8,4608e-07	3,2074e-04	-2,3094e-04
Srednji	7,6196e-03	4,7420e-06	1,4616e-04	2,1250e-04	1,4826e-03	1,7670e-03	-2,2958e-04
Najslabši	2,4920e-02	6,1155e-06	1,6669e-04	4,8226e-02	2,5823e-03	4,4835e-03	1,0198e-01
Povprečje	9,3329e-03 †	4,7780e-06 †	1,4747e-04	2,1184e-02	1,2154e-03 †	1,7640e-03‡	7,9347e-03‡
Standardni odklon	5,8079e-03	5,4239e-07	5,1448e-06	2,4157e-02	7,3065e-04	9,5558e-04	2,8247e-02

Tabela 6. Rezultati za indikator $I_{\overline{H}}$ na testnih funkcijah 8–13 z $M = 3$
Table 6. Results for $I_{\overline{H}}$ indicator on the benchmark functions 8–13 when $M = 3$

	8. S.DTLZ2	9. R.DTLZ2	10. S.DTLZ3	11. WFG1	12. WFG8	13. WFG9
Najboljši	3,7128e-05	9,4432e-03	1,0378e-10	1,9567e-01	-1,6428e-01	-5,7671e-02
Srednji	8,1187e-05	1,1602e-02	1,2057e-08	2,1337e-01	-1,6045e-01	-5,4269e-02
Najslabši	1,3746e-04	1,9873e-02	2,3965e-07	2,2280e-01	-1,5359e-01	-5,0538e-02
Povprečje	8,2895e-05	1,2067e-02 †	3,7420e-08 †	2,1158e-01 †	-1,6011e-01†	-5,4009e-02 †
Standardni odklon	2,8280e-05	2,4892e-03	6,2481e-08	8,1245e-03	2,6295e-03	2,1130e-03

Tabela 7. Rezultati za indikator $I_{\overline{H}}$ na testnih funkcijah 8–13 z $M = 5$
Table 7. Results for $I_{\overline{H}}$ indicator on the benchmark functions 8–13 when $M = 5$

	8. S.DTLZ2	9. R.DTLZ2	10. S.DTLZ3	11. WFG1	12. WFG8	13. WFG9
Najboljši	4,4943e-07	7,5310e-03	3,6717e-10	5,1646e-01	-2,1199e-01	-1,2990e-01
Srednji	1,8934e-06	8,7524e-03	1,6341e-09	5,2159e-01	-1,9635e-01	-1,2384e-01
Najslabši	6,8299e-06	1,1796e-02	6,9752e-09	5,2409e-01	-1,7477e-01	-1,1667e-01
Povprečje	2,6355e-06‡	8,8118e-03 †	2,0630e-09 †	5,2126e-01	-1,9555e-01†	-1,2346e-01 †
Standardni odklon	1,8277e-06	8,8397e-04	1,6132e-09	2,1147e-03	1,0860e-02	3,3397e-03

jskem procesu DE. Mutacijski proces za i -tega mutiranega posameznika $\mathbf{v}_{i,G+1}$ v generaciji $G + 1$ operira takole:

$$\mathbf{v}_{i,G+1} = \mathbf{x}_{r_1,G} + F_{i,G+1} \times (\mathbf{x}_{r_2,G} - \mathbf{x}_{r_3,G}).$$

Proces rekombinacije v algoritmu DEMOWSA je povzet iz strategije 'rand/1/bin', tj. ene temeljnih strategij algoritma DE, zapisanih v [18, 12] oz. prvotnem

DEMO [16]. Krmilni parameter tega procesa, stopnjo križanja CR , smo prilagajali podobno kot krmilni parameter mutacije, stopnjo mutacije F , in zanj izbrali:

$$CR_{i,G+1} = \langle CR_G \rangle_i \times e^{\tau N(0,1)},$$

ter podobno kot prej izračunali:

$$\langle CR_G \rangle_i = (CR_{i,G} + CR_{r_1,G} + CR_{r_2,G} + CR_{r_3,G})/4.$$

Nov $CR_{i,G}$ je nato uporabljen pri izdelavi kandidata $u_{i,j,G+1}$ z binarnim operatorjem križanja:

$$u_{i,j,G+1} = \begin{cases} v_{i,j,G+1} & \text{if } rand(0,1) \leq CR_{i,G+1} \text{ ali} \\ & j = j_{rand} \\ x_{i,j,G} & \text{sicer,} \end{cases}$$

kjer $j \in [1, D]$ označuje j -ti iskalni parameter, $rand(0,1) \in [0,1]$ pomeni vzorčenje uniformno (psevdo) naključno porazdeljenega naključnega števila in j_{rand} pomeni uniformno naključen indeks iskalnega parametra, ki ga vedno vzamemo iz $v_{i,j,G+1}$ (da bi s tem preprečili izdelavo enakih posameznikov).

Po križanju sledi operator selekcije, ki v $G+1$ lahko $x_{i,G}$ zamenja z boljšim $u_{i,G+1}$ in tako pomaga pri ustreznem prilagajanju parametrov F in CR , saj preživijo le posamezniki, ki se prilagajajo dobrim iskalnim in krmilnim parametrom. Zato se iskanje hitreje približa boljšim rešitvam, ki tudi bolj verjetno preživijo, ustvarijo več potomcev in s tem razširijo svoje krmilne parametre [11].

4 Poskusi in rezultati

Za namen študije smo v algoritmu DEMOwSA nastavili naslednje vrednosti parametrov. Krmilna parametra smo omejili na $0,1 \leq F \leq 0,9$ in $1/24 \leq CR \leq 0,9$. Parameter τ smo nastavili na $\tau = 2/\sqrt{2D}$, kar sovпада s priporočili iz [4]. V začetni populaciji imajo vsi posamezniki krmilni parameter $F_{init} = 0,5$ in krmilni parameter $CR_{init} = 0,3$ (predlagan v [16], izbran tudi v [19]). Za študijo smo izbrali selekcijsko strategijo *SPEA2*, ki je dajala najboljše rezultate.

4.1 Indikatorji kakovosti

Predstavljeni algoritem smo 25-krat zagnali na 19 testnih funkcijah [13]. V tabelah 2–7 so najboljša, srednja (medialna), najslabša in povprečna vrednost ter standardna deviacija za indikatorje [14] I_{R2} in I_H (manjša vrednost je boljša) testnih funkcij po 500.000 ovrednotenjih (FES). Iz vseh dobljenih tabel za uporabljene indikatorje je opaziti, da je algoritem v veliki meri uspešen glede na indikatorje. Referenčne množice imajo vrednosti indikatorjev enako 0. Ko je dobljena vrednost indikatorja negativna, aproksimacijska množica presega kakovost podane referenčne množice glede na podan indikator.

4.2 Dinamika krmilnih parametrov F in CR

Iz tabele 1 vidimo povprečja vrednosti krmilnih parametrov F in CR pri vsaki od funkcij po $5e+5$ FES. Posamezna vrednost pomeni povprečje (oz. standardni odklon) zadnjih 25 generacij za vse neodvisne zagone, za primerke, ki so izboljšali starševsko rešitev. Kot vidimo, se krmilna parametra samoprilagodita na precej različne

vrednosti, zato ni mogoče sklepati o fiksni vrednosti posameznega parametra, ki bi enako dobro delovala za vse funkcije. Pri vseh funkcijah vidimo, da začetne vrednosti $CR_{init} = 0,3$ ni med njimi, kar dodatno nakazuje na uspešno samoprilagajanje krmilnih parametrov. Po $5e+5$ FES znaša razpon dobljenih povprečnih vrednosti za parameter F od $0,17 \pm 0,06$ (SYMPART) do $0,68 \pm 0,15$ (R_DTLZ2_M3) ter za parameter CR od $0,06 \pm 0,02$ (S_DTLZ3_M5) do $0,75 \pm 0,16$ (SYMPART). Krmilna parametra F in CR zajameta različne kombinacije nižjih in višjih vrednosti. Tako sta si pri funkciji S_DTLZ2_M5 podobna ter pri funkcijah SYMPART in S_DTLZ3_M5 precej v kontrastu.

Ker je za nekatere izmed funkcij dinamika krmilnih parametrov F in CR precej različna, na slikah 3 in 4 prikazujemo tudi nekaj izbranih grafov le-teh. Iz slik vidimo, da vrednosti pogosto spreminjajo smer samoprilagajanja, saj algoritem menjava faze lokalnega in globalnega iskanja, ki ju krmilna parametra nadzorujeta.

Opravili smo tudi primerjavo med učinkovitostjo samoprilagajanja in fiksnimi krmilnimi parametri F in CR (z enakimi začetnimi nastavitvami za oba poskusa). Iz statističnih t-testov v tabelah 2–7 je razvidno, da je samoprilagajanje za algoritem boljše od fiksnih parametrov. Z † oz. ‡ označimo z 99,9-odstotno zanesljivostjo pomembno izboljšanje oz. poslabšanje pri samoprilagajanju, tj. skupaj 21 izboljšanj indikatorja, 12 poslabšanj in 5 brez razlik. Ker je iz študije dinamike krmilnih parametrov F in CR razvidno, da se ti precej spreminjajo in jih ni mogoče nastaviti enako za poljuben problem, samoprilagajanje še toliko bolj pomaga pri reševanju industrijskih problemov [8].

5 Sklep

V članku smo opravili študijo krmilnih parametrov pri samoprilagodljivem algoritmu diferencialne evolucije DEMOwSA. Z zmožnostjo samoprilagajanja krmilnih parametrov dosega algoritem v povprečju boljše rezultate pri večkriterijski optimizaciji in je bolj robusten za uporabo na novih funkcijah, saj uporabniku ni treba nastaviti krmilnih parametrov F in CR . S tem smo z algoritmom dosegli dva pomembna cilja. Kot nadaljnje delo vidimo izboljšavo mutacijskih in selekcijskih mehanizmov v večkriterijski optimizaciji.

6 Literatura

- [1] H. A. Abbass, R. Sarker in C. Newton. PDE: A Pareto-frontier Differential Evolution Approach for Multi-objective Optimization Problems. V: *Proceedings of the Congress on Evolutionary Computation 2001 (CEC'2001)*, letnik 2, str. 971–978, Piscataway, New Jersey, 2001. IEEE Service Center.
- [2] B. V. Babu in M. M. Leenus Jehan. Differential Evolution for Multi-Objective Optimization. V: *Proceedings of the 2003 Congress on Evolutionary Computation*

- tion (CEC'2003), letnik 4, str. 2696–2703, Canberra, Australia, 2003. IEEE Press.
- [3] T. Bäck. *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*. Oxford University Press, New York, 1996.
- [4] H. G. Beyer in H. P. Schwefel. Evolution strategies — A comprehensive introduction. *Natural Computing*, 1:3–52, 2002.
- [5] J. Brest, B. Bošković, S. Greiner, V. Žumer in M. Sepesy Maučec. Performance comparison of self-adaptive and adaptive differential evolution algorithms. *Soft Computing - A Fusion of Foundations, Methodologies and Applications*, 11(7):617–629, 2007.
- [6] J. Brest, S. Greiner, B. Bošković, M. Mernik in V. Žumer. Self-Adapting Control Parameters V Differential Evolution: A Comparative Study on Numerical Benchmark Problems. *IEEE Transactions on Evolutionary Computation*, 10(6):646–657, 2006.
- [7] J. Brest, V. Žumer in M. S. Maučec. Population size in differential evolution algorithm. *Elektrotehniški vestnik*, 74(1-2):55–60, 2007.
- [8] C. A. Coello Coello. 20 Years of Evolutionary Multi-Objective Optimization: What Has Been Done and What Remains to be Done. V: *Computational Intelligence: Principles and Practice*, poglavje 4, str. 73–88. IEEE Computational Intelligence Soc., Vancouver, Canada, 2006.
- [9] K. Deb. *Multi-Objective Optimization using Evolutionary Algorithms*. John Wiley & Sons, Chichester, UK, 2001. ISBN 0-471-87339-X.
- [10] K. Deb, S. Agrawal, A. Pratab in T. Meyarivan. A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. V: *Proceedings of the Parallel Problem Solving from Nature VI Conference*, str. 849–858, Paris, France, 2000. Springer. Lecture Notes in Computer Science No. 1917.
- [11] Á. E. Eiben, R. Hinterding in Z. Michalewicz. Parameter control in evolutionary algorithms. *IEEE Trans. on Evolutionary Computation*, 3(2):124–141, 1999.
- [12] V. Feoktistov. *Differential Evolution: In Search of Solutions (Springer Optimization and Its Applications)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.
- [13] V. L. Huang, A. K. Qin, K. Deb, E. Zitzler, P. N. Suganthan, J. J. Liang, M. Preuss in S. Huband. Problem Definitions for Performance Assessment & Competition on Multi-objective Optimization Algorithms. Technical Report TR-07-01, Nanyang Technological University et. al., Singapore, 2007.
- [14] J. Knowles, L. Thiele in E. Zitzler. A Tutorial on the Performance Assessment of Stochastic Multiobjective Optimizers. 214, Computer Engineering and Networks Laboratory (TIK), ETH Zurich, Switzerland, February 2006. Revised version.
- [15] K. Liang, X. Yao in C. Newton. Adapting Self-adaptive Parameters in Evolutionary Algorithms. *Applied Intelligence*, 15(3):171–180, 2001.
- [16] T. Robič in B. Filipič. DEMO: Differential Evolution for Multiobjective Optimization. V: *Proceedings of the Third International Conference on Evolutionary Multi-Criterion Optimization – EMO 2005*, letnik 3410 of *Lecture Notes in Computer Science*, str. 520–533. Springer, 2005.
- [17] J. D. Schaffer. Multiple Objective Optimization with Vector Evaluated Genetic Algorithms. V: *Proceedings of the 1st International Conference on Genetic Algorithms*, str. 93–100, Mahwah, NJ, USA, 1985.
- [18] R. Storn in K. Price. Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. *Journal of Global Optimization*, 11:341–359, 1997.
- [19] T. Tušar in B. Filipič. Differential Evolution versus Genetic Algorithms in Multiobjective Optimization. V: *Proceedings of the Fourth International Conference on Evolutionary Multi-Criterion Optimization – EMO 2007*, letnik 4403 of *Lecture Notes in Computer Science*, str. 257–271. Springer, 2007.
- [20] A. Zamuda. Samoprilagajanje krmilnih parametrov pri algoritmu diferencialne evolucije za večkriterijsko optimizacijo. Mag. naloga, Fakulteta za elektrotehniko, računalništvo in informatiko, Univerza v Mariboru, 2008.
- [21] A. Zamuda, J. Brest, B. Bošković in V. Žumer. Differential Evolution for Multiobjective Optimization with Self Adaptation. V: *The 2007 IEEE Congress on Evolutionary Computation CEC 2007*, str. 3617–3624. IEEE Press, 2007.
- [22] E. Zitzler, M. Laumanns in L. Thiele. SPEA2: Improving the Strength Pareto Evolutionary Algorithm for Multiobjective Optimization. V: *Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems*, strani 95–100, Athens, Greece, 2001. International Center for Numerical Methods in Engineering.
- [23] E. Zitzler in S. Künzli. Indicator-Based Selection in Multiobjective Search. V: X. Yao et al., editors, *Parallel Problem Solving from Nature (PPSN VIII)*, str. 832–842, Berlin, Germany, 2004. Springer-Verlag.

Aleš Zamuda je leta 2008 magistriral na podiplomskem študijskem programu računalništvo in informatika na Fakulteti za elektrotehniko, računalništvo in informatiko Univerze v Mariboru. Trenutno je podiplomski študent in zaposlen v Laboratoriju za računalniške arhitekture in jezike, kjer raziskuje na področju večkriterijske optimizacije in proučuje mehanizme samoprilagajanja za diferencialno evolucijo. Je član IEEE Computational Intelligence Society.

Janez Brest je izredni profesor na Fakulteti za elektrotehniko, računalništvo in informatiko Univerze v Mariboru. Zaposlen je v Laboratoriju za računalniške arhitekture in jezike, kjer se ukvarja s paralelnim in porazdeljenim procesiranjem, s programskimi jeziki, ukvarja pa se tudi z optimizacijskimi raziskavami in evolucijskim računanjem s poudarkom na diferencialni evoluciji in samoprilagajanju krmilnih parametrov.

Borko Bošković je diplomiral leta 2003 na Fakulteti za elektrotehniko, računalništvo in informatiko Univerze v Mariboru. Od leta 2000 je zaposlen v Laboratoriju za računalniške arhitekture in jezike, kjer se ukvarja s spletnim programiranjem, programskimi jeziki, evolucijskimi algoritmi in iskalnimi algoritmi s poudarkom na algoritmih za igranje iger med dvema igralcema, popolno informacijo in ničelno vsoto.