Večkriterijska optimizacija: primerjava algoritmov *MOjDE* in *DEMO*

Janez Brest, Aleš Zamuda, Borko Bošković, Viljem Žumer
Laboratorij za računalniške arhitekture in jezike
Inštitut za računalništvo
Fakulteta za elektrotehniko, računalništvo in informatiko, Univerza v Mariboru
Smetanova 17, 2000 Maribor, Slovenija
E-pošta: janez.brest@uni-mb.si

Multiobjective optimization: Comparison of *MOjDE* and *DEMO* Algorithms

Differential Evolution (DE) is a simple yet powerful evolutionary algorithm for global optimization. $DEMO$ is well-known algorithm for multiobjective optimization. This paper presents empirical results on a set of (CEC'2007) benchmark functions. The results were obtained by our self-adaptive differential evolution algorithm, called $MOjDE$, which is an extension of $DEMO$ algorithm with self-adaptive control parameters, first proposed by J. Brest et al. [3]. The paper outlines that the self-adative algorithm $MOjDE$ gives better performance results than $DEMO$.

1 Uvod

V članku obravnavamo večkriterijsko optimizacijo. Posebej se posvetimo znanemu algoritmu $DEMO$ [19, 23] za večkriterijsko optimizacijo. Algoritem $DEMO$ smo nadgradili s konceptom samo-prilagodljivih kontrolnih parametrov [3]. Novo nastali algoritem imenujmo $MOjDE$. V članku bomo opisali primerjavo algoritmov $DEMO$ in $MOjDE$. Primerjavo smo izvedli na testnih funkcijah in s pomočjo orodij, ki so bila predlagana na posebni sekciji za oceno učinkovitosti večkriterijskih algoritmov za optimizacijo IEEE Congress on Evolutionary Computation (CEC2007) [14].

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, pride do tega, da izboljšanje ocene rešitve po enem kriteriju povzroči poslabšanje ocene po drugih kriterijih. V takšnem primeru nimamo opravka samo z eno optimalno rešitvijo, temveč z množico optimalnih rešitev, zbranih v *Pareto optimalni fronti*. Za izbiro najugodnejših rešitev problema pogosto želimo dobro spoznati, kakšne možnosti sploh imamo, oz. kakšne najboljše (Pareto optimalne) možnosti imamo ter se šele kasneje odločiti za eno od teh. Slednji pristop opisuje večkriterijsko optimizacijo, t.j. sočasno optimizacijo medsebojno konfliktnih kriterijev.

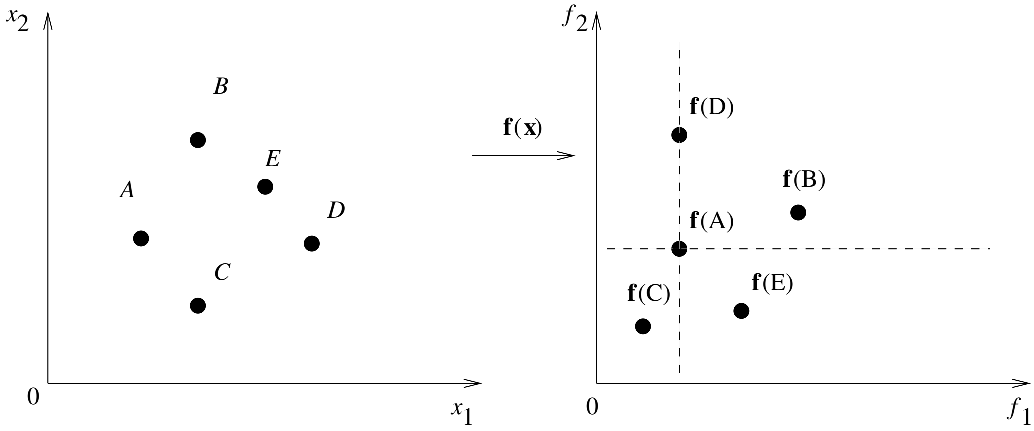
V začetku večkriterijske optimizacije so se uporabljale klasične enokriterijske optimizacijske metode, pri tem se je večkriterijski problem z utežno funkcijo preslikal v enokriterijski problem. Prvi poskus reševanja večkriterijskega problema z iskanjem vseh možnih rešitev je bil predstavljen v [20]. Od takrat naprej [5], še najbolj pa v zadnjem desetletju se je pri večkriterijski optimizaciji precej povečal interes za uporabo populacijskih naključnih iskalnih algoritmov [7, 8, 26, 1, 27, 15, 19, 9], kot so evolucijski algoritmi, simulirano ohlajanje in iskanje s tabuji. S temi želimo najti čim več in čim pravilneje porazdeljenih Pareto optimalnih rešitev. Evolucijski algoritmi (EA) so se pokazali kot zelo uporabni pri reševanju kompleksnih večkriterijskih optimizacijskih problemov, vključno s številnimi problemi iz dejanskega sveta. V ta namen se je pojavilo veliko posebnih iskalnih algoritmov, njihova uporabnost pa je bila prikazana v različnih domenah [7]. Primeri večkriterijskih optimizacijskih problemov iz dejanskega sveta so npr.: optimizacija stacionarne plinske gorilne turbine [4], oblikovanje drobilca kamnov [2], distribucija surovin skozi mrežo oljnih kanalov [6], upravljanje z jedrskim gorivom [10], planiranje [21], načrtovanje telekomunikacijskih omrežij [24], obrambne aplikacije [16] itd. Za ocenitev učinkovitosti iskalnih algoritmov so bile predlagane številne metrike [17, 13].

Preostanek prispevka je organiziran takole. V drugem poglavju formalno predstavimo problem večkriterijske optimizacije in razložimo osnovne izraze, ki so z njo povezani. Tretje poglavje govori o pristopih za ocenjevanje in primerjavo algoritmov za večkriterijsko optimizacijo. V četrtem poglavju predstavljamo nastavitev parametrov in eksperimentalne rezultate pri večkriterijski optimizaciji na izbranih testnih funkcijah. Sledi še zaključno poglavje z idejami za nadaljnje delo.

2 Večkriterijska optimizacija

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

$$\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_M(\mathbf{x}))$$



Slika 1: Pojem dominantnosti na primeru večkriterijske funkcije $\mathbf{f}(\mathbf{x})$.

in spremenljivke zadoščajo m neenakostnim omejitvam:

$$g_i(\mathbf{x}) \geq 0; \quad i = 1, 2, ..., m.$$

Rešitev večkriterijskega optimizacijskega problema $\mathbf{x}$ *dominira* rešitev $\mathbf{y}(\mathbf{x} \preceq \mathbf{y})$, če velja:

1. Rešitev $\mathbf{x}$ ni slabša od rešitve $\mathbf{y}$ po vseh kriterijih ($\mathbf{f}_i(\mathbf{x}) \leq \mathbf{f}_i(\mathbf{y}), \forall i = 1, ..., k$)).

2. Rešitev $\mathbf{x}$ je boljša od rešitve $\mathbf{y}$ po vsaj enem kriteriju ($\exists j \in \{1, ..., k\}$: $\mathbf{f}_j(\mathbf{x}) < \mathbf{f}_j(\mathbf{y})$).

Prostor kriterijev $\Re^k$ (k 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 C najboljša izmed vseh ($\mathbf{f}(\mathbf{C})$ je manjša od ostalih po vseh kriterijih), rešitev A pa je boljša od B in D ter neprimerljiva z E.

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 *Pareto optimalna fronta*, njeni elementi pa *Pareto optimalne rešitve*[1]. Na sliki 1 tvori množico nedominiranih rešitev le ena rešitev, to je C.

Pri večkriterijski optimizaciji sta ponavadi pomembna naslednja cilja: 1) (enakomerna) porazdelitev rešitev po Pareto čelu in 2) bližina k Pareto čelu.

3 Metrike učinkovitosti

Metrike učinkovitosti oz. *indikatorji kvalitete* (za razliko med pojmoma glej [17], str. 6) numerično izmerijo uspeh *aproksimacijske množice* (approximation set) nedominiranih rešitev. Metrike kažejo zgolj videnje ocenjevalca, kar pomeni, da sta lahko dve rešitvi z različnimi metrikami popolnoma različno ocenjeni oz. relativno razvrščeni glede na preostale rešitve.

Metrike učinkovitosti se ponavadi računajo glede na podano referenčno aproksimacijsko množico. S primerjavo izračunane aproksimacijske množice in referenčne aproksimacijske množice ugotovimo, katera je boljša. Poleg referenčne aproksimacijske množice pa pogosto potrebujemo tudi referenčno točko, ki navzgor omejuje vse rešitve, t.j. je slabša po vsakem od kriterijev od vseh rešitev v vsaki naši aproksimacijski množici. To je najlažje izvesti tako, da dobljene vrednosti kriterijev v rešitvah normaliziramo na interval $[1, 2]$, referenčno točko pa postavimo na vrednost 2,1 za vsak kriterij.

Najbolj znan indikator kvalitete je hipervolumen (indikator H) [28]. Ta meri dominiran prostor rešitev, ki ga je algoritem že našel. Metrika deluje nad diskretnim prostorom rešitev, tako da zvezen prostor rešitev diskretizira v diskretni prostor. Ta indikator meri učinkovitost obeh ciljev večkriterijske optimizacije, tako porazdelitev rešitev po Pareto čelu, kot tudi bližino k Pareto čelu.

Indikator R [12] meri zgolj oddaljenost od Pareto čela, pri tem pa vsako večkriterijsko rešitev pretvori v realno vrednost. Pri pretvorbi upošteva pomožno funkcijo za utežitev kriterijev (npr. funkcijo Čebišova).

Poleg numeričnih metrik učinkovitosti je pogosto uporabna tudi vizualizacija dobljenih aproksimacijskih množic. Ena od takšnih metod je izris (npr. 50%) empirično doseženih ploskev. To ploskev dobimo tako, da na diskretnem prostoru rešitev ugotovimo, katere rešitve je iskalni algoritem dosegel (npr. v 50%).

4 Eksperimenti in rezultati

4.1 Nastavitev parametrov pri algoritmu $MOjDE$

Algoritem $MOjDE$ uporablja DE strategijo *DE/rand/1/bin* [22, 18, 11].

V generaciji G za posameznika $\mathbf{x}_i$ izračunamo nova kontrolna parametra $F_{i,G+1}$ in $CR_{i,G+1}$ takole [3]:

$$F_{i,G+1} = \begin{cases} F_l + rand_1 * F_u & \text{if } rand_2 < \tau_1, \\ F_{i,G} & \text{sicer,} \end{cases}$$

$$CR_{i,G+1} = \begin{cases} CR_l + rand_3 * CR_u & \text{if } rand_4 < \tau_2, \\ CR_{i,G} & \text{sicer.} \end{cases}$$

[1]Po Vilfredu Paretu, 1848 – 1923.

$rand_j, j \in \{1,2,3,4\}$ so naključne realne vrednosti na intervalu $[0,1)$. τ_1 in τ_2 sta verjetnosti, s katerima uravnavamo kontrolna parametra F in CR. Konstante imajo naslednje vrednosti: $\tau_1 = 0,1$; $\tau_2 = 0,1$; $F_l = 0,1$; $F_u = 0,9$; $CR_l = 0,1$ in $CR_u = 0,6$. Nov F ima naključno vrednost na intervalu $[0,1;1,0)$, nov CR pa na intervalu $[0,1;0,7)$. Vrednosti $F_{i,G+1}$ in $CR_{i,G+1}$ sta izračunani pred mutacijo in vplivata na mutacijo, križanje in selekcijo novega posameznika $\mathbf{x}_{i,G+1}$.

Pravkar opisan izračun samo-prilagodljivih kontolnih parametrov F in CR predstavlja (pomembno) razliko med algoritmoma $MOjDE$ in $DEMO$.

V začetni populaciji imajo vsi posamezniki kontrolni parameter $F_{init} = 0,5$ in kontrolni parameter $CR_{init} = 0,7$.

4.2 Nastavitev parametrov pri algoritmu $DEMO$

Algoritem $DEMO$ ne spreminja vrednosti kontrolnih parametrov med evulicijskim postokov in fiksne vrednosti kontrolnih parametrov so: $F = 0,5$ in $CR = 0,3$. $DEMO$ ima podporo za več selekcijskih strategij ($NSGA$-II, $SPEA$, $IBEA$). Med njimi po naših raziskavah daje najboljše rezultate selekcijska strategija $SPEA$, ki smo jo uporabili pri obeh algoritmih v tem članku.

4.3 Rezultati

Rezultati poskusov na testnih funkcijah CEC'2007 so prikazani v tabelah 1–6 (v članku [25]), kjer vrednost v tabeli predstavlja razliko med vrednostjo dobljeno z algoritmom $MOjDE$ in vrednostjo dobljeno z algoritmom $DEMO$. Predstavitev z razliko nam omogoča lažjo primerjavo obeh algoritmov, saj v primeru pozitivne vrednosti to pomeni, da je algoritem $MOjDE$ boljši v primerjavi z algoritmom $DEMO$. Rezultati v tabelah so prikazani pri treh različnih številih evaluacij funkcij (FES): 5.000, 50.000 in 500.000 ($5e+5$), kjer zadnja vrednost predstavlja tudi najpomembnejši rezultat. Vseh zagonov za posamezno funkcijo je bilo 25. V tabeli vidimo najboljši rezultat (Best), mediano in najslabši rezultat ter povprečno vrednost in standardni odklon.

Pri povprečnih vrednostih pri indikatorju R vidimo, da je algoritem $MOjDE$ pri $5e+5$ FES 14-krat boljši in 5-krat slabši v primerjavi z $DEMO$. Če pa opravimo primerjavo s t-testom (pri 99,9% zanesljivosti), ugotovimo, da je $MOjDE$ v poprečju 13-krat (signifikantno) boljši, 4-krat slabši (S_ZDT2, R_ZDT4, S_ZDT6 in S_ZDTZ2 pri $M = 5$) in 2-krat t-test ne kaže signifikantne razlike.

Pri povprečnih vrednostih pri indikatorju H vidimo, da algoritem $MOjDE$ pri $5e+5$ FES pravtako 14-krat boljši in 5-krat slabši v primerjavi z $DEMO$. Če pa opravimo primerjavo s t-testom (pri 99,9% zanesljivosti), ugotovimo, da je $MOjDE$ v poprečju 13-krat (signifikantno) boljši, 4-krat slabši (S_ZDT2, R_ZDT4, S_ZDT6 in S_ZDTZ2 pri $M = 5$) in 2-krat t-test ne kaže signifikantne razlike.

Do podobnih ugotovitev pridemo, če opravimo primerjavo najboljših rezultatov večkriterijske optimizacije.

V članku [25] so prikazane slike empirično 0%, 50% in 100% desegljive ploskev za prvih 7 testnih funkcij.

5 Zaključek

V članku smo opravili primerjavo dveh evolucijskih algoritmov za večkriterijsko optimizacijo. Oba algoritma temeljita na differencialni evoluciji. Bistvena razlika med njima je v uporabi samo-prilagodljivih kontrolnih parametrov diferencilne evolucije pri algoritmu $MOjDE$. Eksperimentalni rezultati na testnih funkcijah (CEC'07) za večkriterijsko optimizacijo kažejo, da algoritem $MOjDE$ v popračju daje boljše rezultate kot algoritem $DEMO$, ki ima fiksne kontrolne parametre.

Kot nadaljenje delo pri obeh algoritmih omenimo uporabo lokalne optimizacije, ki bi lahko izboljšala učinkovitost obeh v tem članku predstavljenih algoritmov za večkriterijsko optimizacijo.

Literatura

[1] Hussein A. Abbass, Ruhul Sarker, and Charles Newton. PDE: A Pareto-frontier Differential Evolution Approach for Multi-objective Optimization Problems. In *Proceedings of the Congress on Evolutionary Computation 2001 (CEC'2001)*, volume 2, pages 971–978, Piscataway, New Jersey, May 2001. IEEE Service Center.

[2] L. Barone, L. While, and P. Hingston. Designing Crushers with a Multi-Objective Evolutionary Algorithm. In W.B. Langdon, E. Cantú-Paz, K. Mathias, R. Roy, D. Davis, R. Poli, K. Balakrishnan, V. Honavar, G. Rudolph, J. Wegener, L. Bull, M.A. Potter, A.C. Schultz, J.F. Miller, E. Burke, and N. Jonoska, editors, *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'2002)*, pages 995–1002, San Francisco, California, July 2002. Morgan Kaufmann Publishers.

[3] 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. *IEEE Transactions on Evolutionary Computation*, 10(6):646–657, 2006. DOI: 10.1109/TEVC.2006.872133.

[4] Dirk Büche, Peter Stoll, Rolf Dornberger, and Petros Koumoursakos. Multiobjective Evolutionary Algorithm for the Optimization of Noisy Combustion Processes. *IEEE Transactions on Systems, Man, and Cybernetics Part C—Applications and Reviews*, 32(4):460–473, November 2002.

[5] Carlos A. Coello Coello. 20 Years of Evolutionary Multi-Objective Optimization: What Has Been Done and What Remains to be Done. In Gary Y. Yen and David B. Fogel, editors, *Computational Intelligence: Principles and Practice*, chapter 4, pages 73–88. IEEE Computational Intelligence Society, Vancouver, Canada, 2006, ISBN 0-9787135-0-8.

[6] J.M. de la Cruz, B. de Andres-Toro, A. Herrán, E. Besada Porta, and P. Fernandez Blanco. Multiobjective optimization of the transport in oil pipelines. In *Proceedings of the*

9th IEEE International Conference on Emerging Technologies and Factory Automation, volume 1, pages 566–573, Lisbon, Portugal, September 2003. IEEE.

[7] Kalyanmoy Deb. *Multi-Objective Optimization using Evolutionary Algorithms*. John Wiley & Sons, Chichester, UK, 2001. ISBN 0-471-87339-X.

[8] Kalyanmoy Deb, Samir Agrawal, Amrit Pratab, and T. Meyarivan. A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. In Marc Schoenauer, Kalyanmoy Deb, Günter Rudolph, Xin Yao, Evelyne Lutton, J. J. Merelo, and Hans-Paul Schwefel, editors, *Proceedings of the Parallel Problem Solving from Nature VI Conference*, pages 849–858, Paris, France, 2000. Springer. Lecture Notes in Computer Science No. 1917.

[9] Francesco di Pierro and Soon-Thiam Khu an Dragan A. Savić. An Investigation on Preference Order—Ranking Scheme for Multiobjective Evolutionary Optimization. *IEEE Transactions on Evolutionary Computation*, 11(1):17–45, 2007. DOI: 10.1109/TEVC.2006.876362.

[10] P. Engrand. A multi-objective optimization approach based on simulated annealing and its application to nuclear fuel management. In *Proceedings of the Fifth International Conference on Nuclear Engineering*, pages 416–423, Nice, France, May 1997. American Society of Mechanical Engineering.

[11] Vitaliy Feoktistov. *Differential Evolution: In Search of Solutions (Springer Optimization and Its Applications)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[12] M. Hansen and A. Jaskiewicz. Evaluating the Quality of Approximations to the Non-dominated Set. Technical Report IMM-REP-1998-7, Institue of Mathematical Modelling, Technical University of Denmark, 1998.

[13] Michael Pilegaard Hansen and Andrzej Jaszkiewicz. Evaluating the quality of approximations to the non-dominated set. Technical Report IMM-REP-1998-7, Technical University of Denmark, March 1998.

[14] V. L. Huang, A. K. Qin, K. Deb, E. Zitzler, P. N. Suganthan, J. J. Liang, M. Preuss, and 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.

[15] V. L. Huang, P. N. Suganthan, A. K. Qin, and S. Baskar. Multiobjective Differential Evolution with External Archive and Harmonic Distance-Based Diversity Measure. Technical Report TR-07-01, Nanyang Technological University, Singapore, 2006.

[16] Evan J. Hughes. Swarm Guidance using a Multi-Objective Co-evolutionary On-Line Evolutionary Algorithm. In *2004 Congress on Evolutionary Computation (CEC'2004)*, volume 2, pages 2357–2363, Portland, Oregon, USA, June 2004. IEEE Service Center.

[17] J. Knowles, L. Thiele, and 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.

[18] K. V. Price, R. M. Storn, and J. A. Lampinen. *Differential Evolution, A Practical Approach to Global Optimization*. Springer, 2005.

[19] Tea Robič and Bogdan Filipič. DEMO: Differential Evolution for Multiobjective Optimization. In *Proceedings of the Third International Conference on Evolutionary Multi-Criterion Optimization – EMO 2005*, volume 3410 of *Lecture Notes in Computer Science*, pages 520–533. Springer, 2005.

[20] J. David Schaffer. Multiple Objective Optimization with Vector Evaluated Genetic Algorithms. In *Proceedings of the 1st International Conference on Genetic Algorithms*, pages 93–100, Mahwah, NJ, USA, 1985. Lawrence Erlbaum Associates, Inc.

[21] K. J. Shaw, A. L. Nortcliffe, M. Thompson, J. Love, C. M. Fonseca, and P. J. Fleming. Assessing the Performance of Multiobjective Genetic Algorithms for Optimization of a Batch Process Scheduling Problem. In *1999 Congress on Evolutionary Computation*, pages 37–45, Washington, D.C., July 1999. IEEE Service Center.

[22] R. Storn and K. Price. Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. *Journal of Global Optimization*, 11:341–359, 1997.

[23] Tea Tušar and Bogdan Filipič. Differential Evolution versus Genetic Algorithms in Multiobjective Optimization. In *Proceedings of the Fourth International Conference on Evolutionary Multi-Criterion Optimization – EMO 2007*, volume 4403 of *Lecture Notes in Computer Science*, pages 257–271. Springer, 2007.

[24] S. Watanabe, T. Hiroyasu, and M. Miki. Parallel Evolutionary Multi-Criterion Optimization for Mobile Telecommunication Networks Optimization. In K.C. Giannakoglou, D.T. Tsahalis, J. Periaux, K.D. Papailiou, and T. Fogarty, editors, *Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems. Proceedings of the EUROGEN'2001. Athens. Greece, September 19-21*, pages 167–172, Baracelona, Spain, 2001. International Center for Numerical Methods in Engineering(CIMNE).

[25] Aleš, Zamuda, Janez Brest, Borko Boškovič, Viljem Žumer. Večkriterijska optimizacija: eksperimentalni rezultati algoritmov *MOjDE* in *DEMO* In Baldomir Zajc, urednik, *Zbornik šestnajste mednarodne Elektrotehniške in računalniške konference ERK 2007*, poslano.

[26] E. Zitzler, M. Laumanns, and L. Thiele. SPEA2: Improving the Strength Pareto Evolutionary Algorithm for Multiobjective Optimization. In K. C. Giannakoglou, D. T. Tsahalis, J. Périaux, K. D. Papailiou, and T. Fogarty, editors, *Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems*, pages 95–100, Athens, Greece, 2001. International Center for Numerical Methods in Engineering (Cmine).

[27] Eckart Zitzler and Simon Künzli. Indicator-Based Selection in Multiobjective Search. In Xin Yao et al., editors, *Parallel Problem Solving from Nature (PPSN VIII)*, pages 832–842, Berlin, Germany, 2004. Springer-Verlag.

[28] Eckart Zitzler and Lothar Thiele. Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach. *IEEE Transactions on Evolutionary Computation*, 3(4):257–271, 1999.