

Uglaševanje parametrov evolucijskega algoritma za večkriterijsko optimiranje na industrijskem problemu

Matjaž Depolli¹, Tea Tušar², Bogdan Filipič²

¹Odsek za komunikacijske sisteme

²Odsek za inteligentne sisteme

Institut "Jožef Stefan"

Jamova 39, 1000 Ljubljana

matjaz.depolli@ijs.si, tea.tusar@ijs.si, bogdan.filipic@ijs.si

Tuning Parameters of a Multiobjective Optimization Evolutionary Algorithm on an Industrial Problem

We present an empirical study of tuning parameters of an evolutionary algorithm on the steel casting optimization problem. The optimization task consists of setting 18 coolant flows in such a way, that the following two criteria are optimized: deviation of the steel core length from the target length and the sum of deviations of steel temperatures from the target temperatures. Because the two objectives are conflicting, more than one optimal solution exists. A computer simulator of the casting process is used to evaluate solutions that are generated with the algorithm DEMO (Differential Evolution for Multiobjective Optimization). The most limiting factor for the optimization is the time needed for the evaluation of solutions. Therefore we limit the total number of evaluations and search for the best parameter setting of DEMO. The parameters considered are the population size, crossover probability and scaling factor. The results show that population size has a great influence on the performance of the algorithm, while the settings of crossover probability and scaling factor are not crucial for performance of DEMO on this task.

1. Uvod

Najpomembnejša razlika med enokriterijskim in večkriterijskim optimiranjem je v številu kriterijev, ki določajo optimalnost rešitve. Druga razlika izhaja iz prve in se pojavi, ko si kriteriji večkriterijskega optimiranja nasprotujejo: takrat optimalna rešitev ni ena sama, ampak jih je več. Različni kompromisi med optimalnostjo posameznih kriterijev namreč določajo različne rešitve. Šele z upoštevanjem dodatnega znanja lahko kriterijem določimo pomembnost in izmed množice optimalnih rešitev izberemo eno samo najboljšo rešitev.

Optimalnost rešitev v večkriterijski optimizaciji določamo po načelu Pareto optimalnosti. Nedominirane rešitve (rešitve, za katere ne obstaja nobena druga rešitev, ki bi bila po vseh kriterijih boljša ali enako dobra) tvorijo Pareto optimalno fronto. Cilj algoritmov za večkriterijsko optimizacijo je najti množico nedominiranih rešitev, ki so čim bližje Pareto optimalni fronti in so čim bolj raznolike.

Iskanje množice rešitev je značilnost in hkrati tudi posebnost evolucijskih algoritmov, zato jih tudi najpogosteje uporabljamo za večkriterijsko optimizacijo [1, 7]. Primer je algoritem DEMO [6], ki omogoča večkriterijsko optimiranje na podlagi diferencialne evolucije. Diferencialna evolucija je novejši algoritem iz družine evolucijskih algoritmov, ki je bil razvit za reševanje numeričnih optimizacijskih problemov [5].

Primer optimiranja numeričnega problema, ki ga obravnavamo v tem prispevku, je iskanje nastavitev procesnih parametrov v kontinuiranem ulivanju jekla [3]. Gre za nastavljanje intenzivnosti hlajenja jekla na različnih točkah v procesu, ki vpliva na kakovost jekla, porabo hladila in varnost proizvodnega postopka. Naše izkušnje iz enokriterijskega optimiranja tega procesa v železarnah Acroni, Jesenice [2] in Ruukki Steel, Raahe, Finska [3] tokrat nadgrajujemo z večkriterijskim optimiranjem. Obravnavamo problem iz finske železarne, s tem da upoštevamo dva kriterija: vsoto odstopanj temperatur jekla od ciljnih v osemnajstih točkah v livni napravi in odstopanje dolžine nestrjenega dela jekla od ciljne dolžine. Po udarek dajemo na uglaševanje parametrov optimizacijskega algoritma.

V nadaljevanju članka na kratko opišemo industrijski postopek, ki ga želimo optimirati, in cilje optimizacije. Nadaljujemo z opisom parametrov algoritma DEMO, ki jih uglašujemo (velikost populacije, verjetnost križanja in faktor skaliranja), opisom metrik (hipervolumen in površina dosega), s pomočjo katerih se odločamo o primernosti parametrov algoritma, ter načrtom poskusov. Nazadnje podamo rezultate v grafični obliki ter jih komentiramo.

2. Večkriterijsko optimiranje kontinuiranega ulivanja jekla

Kontinuirano ulivanje je proces, ki se zaradi svoje učinkovitosti in ekonomičnosti uporablja za proizvodnjo jekla. Staljeno jeklo se uliva v kokilo, ki ga delno ohladi in oblikuje v slab. Jeklo nato potuje skozi livno napravo, kjer hladilne prhe njegovo temperaturo postopoma znižujejo do te mere, da je celoten presek slaba strjen in ga je mogoče varno odrezati. Postopno hlajenje je skrbno načrtovano in odstopanja od predpisanih ciljnih temperatur pomenijo manjšo kakovost proizvedenega jekla.

Pomemben element v tem procesu so prhe, ki pršijo hladilo (vodo) na jeklo, ki izstopa iz kokile. V obravnavanem primeru se intenzivnost hlajenja nastavlja z osemnajstimi pretoki hladila. Naloga optimizacije je nastavitve vseh pretokov tako, da bosta minimizirana dva kriterija. Prvi je vsota odstopanj temperatur jekla od ciljnih temperatur v osemnajstih točkah v livni napravi. Drugi kriterij je odstopanje dolžine nestrjenega dela jekla od ciljne dolžine. Izkaže se, da si kriterija nasprotujeta – minimizacija enega kriterija otežuje minimizacijo drugega.

Rešitve problema so podane v obliki numeričnega vektorja osemnajstih nastavitvev pretokov, katerih preslikava v prostor kriterijev ni trivialna. Proces hlajenja je potrebno računalniško simulirati. Vrednotenje ene rešitve traja približno 40 sekund na računalniku AMD Athlon™ 64 3200+. Ta časovna zahtevnost omejuje število vrednotenj za evolucijski algoritem. Zato je optimalna izbira parametrov algoritma zelo pomembna.

3. Numerični poskusi

Ker so možnosti eksperimentiranja omejene s časovno zahtevnostjo vrednotenja rešitev, smo želeli z uglaševanjem parametrov algoritma DEMO poleg bližine rešitev Pareto optimalni fronti in njihove razpršenosti zagotoviti tudi hitro konvergenco množice nedominiranih rešitev k Pareto optimalni fronti. Pri konstantnem številu vrednotenj smo iskali optimalne vrednosti za tri parametre algoritma DEMO, ki so hkrati parametri diferencialne evolucije s shemo DE/1/rand/bin [5]: velikost populacije, faktor skaliranja in verjetnost križanja.

Velikost populacije je število rešitev, ki jih algoritem optimira v eni generaciji. To število hkrati določa tudi največjo možno velikost množice nedominiranih rešitev, ki jo dobimo v neki generaciji. S shemo DE/1/rand/bin se kandidat za novo rešitev izračuna z linearno kombinacijo treh obstoječih rešitev. Pri tem se kot parameter uporablja *faktor skaliranja*, ki je navadno v intervalu $(0, 1]$. *Verjetnost križanja* pri diferencialni evoluciji nima enakega pomena kot pri

običajnih evolucijskih algoritmi. Pri križanju kandidata in njegovega starša se vedno ohrani vsaj ena koordinata kandidata. Verjetnost križanja določa verjetnost, s katero se ohrani vsaka preostala koordinata kandidata.

Hitrost konvergence ter kvaliteto množic rešitev smo merili ter primerjali s pomočjo hipervolumna [9, 8] in površine dosega [4]. *Hipervolumen* izračuna hipervolumen prostora, ki ga omejujejo nedominirane rešitve ene generacije na eni strani in referenčna točka (ki je slabša od najslabše možne rešitve) na drugi strani. Večji kot je hipervolumen, boljše je ocenjevana generacija. Referenčna točka je bila določena eksperimentalno in je bila enaka za vse opravljene poskuse. Medtem ko hipervolumen dobro služi za spremljanje napredka optimizacije, je *površina dosega* uporabna predvsem za vizualizacijo dobljenih front. Ko imamo na voljo nedominirane fronte, pridobljene pri različnih nastavitvah parametrov algoritma, jih s površino dosega lahko "povprečimo". Tako npr. 50-odstotna površina dosega prikazuje, kateri del prostora je bil dominiran v vsaj 50% nedominiranih front.

Na podlagi izkušenj z algoritmom DEMO in obravnavanim problemom smo določili naslednjo izhodiščno nastavitve parametrov:

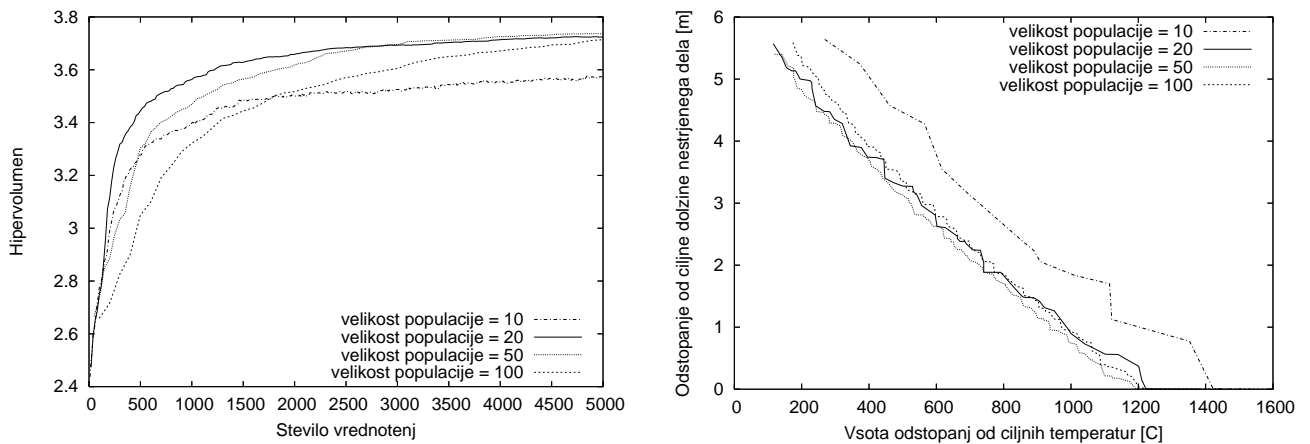
- število vrednotenj $N = 5000$,
- velikost populacije $n = 50$,
- faktor skaliranja $F = 0.5$,
- verjetnost križanja $c = 0.1$.

Nato smo predpostavili, da so medsebojne odvisnosti med naštetimi parametri dovolj šibke, da jih lahko preizkušamo vsakega posebej. V seriji poskusov smo posamično spreminjali vrednosti parametrov n , F in c , pri čemer smo pri vsakem poskusu obdržali izhodiščne nastavitve ostalih parametrov.

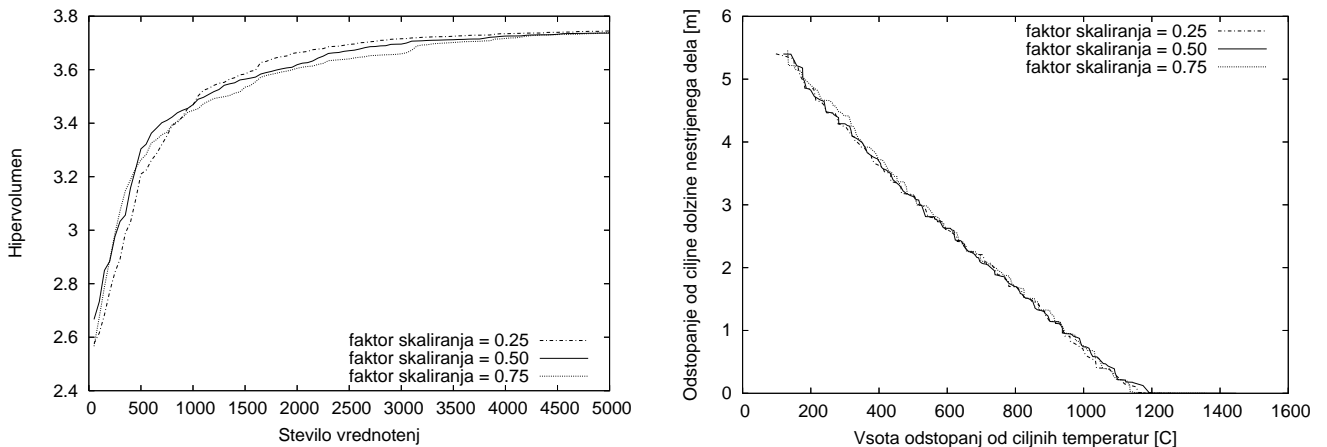
4. Rezultati

Vse poskuse smo ponovili petkrat. Na slikah 1 do 3 so prikazani povprečni hipervolumni in 50-odstotne površine dosega rešitev poskusov.

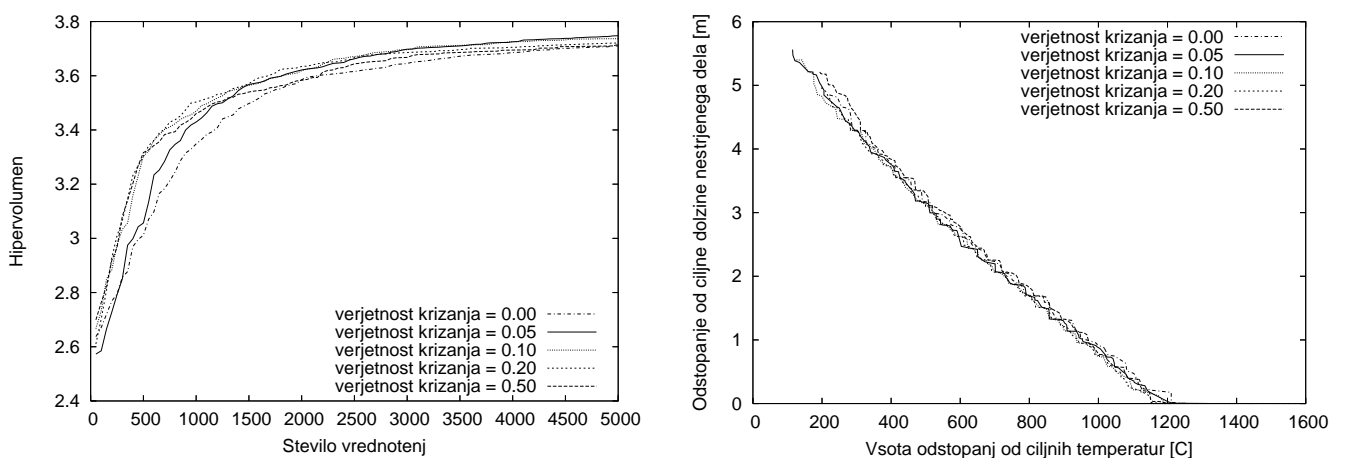
Najprej smo preizkusili reševanje problema s populacijami velikimi 10, 20, 50 in 100 osebkom. Ker je metrika hipervolumna pristranska za različne velikosti populacije (izmed dveh populacij, ki sta enakomerno razporejeni na isti razdalji od Pareto optimalne fronte bo hipervolumen večje populacije večji od hipervolumna manjše populacije), je treba grafe hipervolumna (slika 1 levo) jemati s pridržkom. Iz njih razberemo le hitrost konvergence, medtem ko iz 50-odstotnih površin dosega zadnje fronte (slika 1 desno) razberemo katere rešitve so boljše. Najbolje



Slika 1: Hipervolumen populacij (levo) in 50-odstotna površina dosega zadnjih front (desno) v poskusih z različno velikostjo populacije pri verjetnosti križanja 0.10 in faktorju skaliranja 0.50.



Slika 2: Hipervolumen populacij (levo) in 50-odstotna površina dosega zadnjih front (desno) v poskusih z različnim faktorjem skaliranja pri velikosti populacije 50 in verjetnosti križanja 0.10.



Slika 3: Hipervolumen populacij (levo) in 50-odstotna površina dosega zadnjih front (desno) v poskusih z različno verjetnostjo križanja pri velikosti populacije 50 in faktorju skaliranja 0.50.

sta se izkazali populaciji z 20 in 50 osebki. Druga je kljub počasnejši konvergenci dosegla boljše rešitve. Populacija z 10 osebki je konvergirala hitro, a v dosti slabše rešitve od ostalih, medtem ko je populacija s 100 osebki konvergirala prepočasi. Vsekakor lahko zaključimo, da je pri omejitvi na 5000 vrednotenj populacija velikosti 50 najboljša izmed preverjenih. Če bi imeli na voljo večje število vrednotenj, bi bilo koristno poskusiti z večjo populacijo.

Sledili so poskusi z različnimi vrednostmi faktorja skaliranja (slika 2): 0.25, 0.5 in 0.75. Glede na 50-odstotno površino dosega zadnje fronte so se vse tri preizkušene vrednosti po 5000 vrednotenjih izkazale za enakovredne. Tudi iz grafa hipervolumna je zaznati le rahlo hitrejšo konvergenco pri manjših vrednostih faktorja skaliranja. Optimiranje pretokov hladila z algoritmom DEMO je torej razmeroma neobčutljivo na faktor skaliranja.

Nazadnje smo preizkusili še pet različnih verjetnosti križanja: 0.5, 0.2, 0.1, 0.05 in 0. Verjetnosti večjih od 0.5 nismo preizkušali, saj so se manjše verjetnosti izkazale za boljše. Ob upoštevanju zgolj grafa površine dosega (slika 3 desno) se je težko odločiti, katera verjetnost daje boljše rezultate. Graf hipervolumna (slika 3 levo) pa pokaže, da je vsaka manjša verjetnost križanja dala boljše rezultate, a le do verjetnosti 0.05. Če verjetnost križanja postavimo na 0, so rezultati spet slabši. Vendar pa so razlike med posameznimi rezultati zelo majhne in spreminjanje verjetnosti križanja na rezultate ne vpliva toliko kot spreminjanje velikosti populacije.

5. Zaključek

Na primeru optimizacije pretokov hladila v kontinuiranem ulivanju jekla smo pokazali, da je velikost populacije najbolj vpliven parameter za prilagajanje optimizacijskega algoritma za večkriterijsko optimiranje DEMO. Manjšanje populacije namreč poskrbi za hitrejšo konvergenco k Pareto optimalni fronti, kar smo s pridom izkoristili, saj nam je časovna zahtevnost vrednotenja rešitev postavila omejitev na skupno število vrednotenj. Pokazali smo tudi, da obstaja spodnja meja velikosti populacije, pod katero nadaljnje krčenje populacije poslabša rezultate. Ostala dva parametra (faktor skaliranja in verjetnost križanja) ne vplivata bistveno na učinkovitost algoritma, zato lahko zanj privzamemo vrednosti, ki so se dobro izkazale na podobnih problemih.

Literatura

- [1] K. Deb. *Multi-Objective Optimization using Evolutionary Algorithms*. John Wiley & Sons, Chichester, UK, 2001.
- [2] B. Filipič. Efficient simulation-based optimization of process parameters in continuous casting of steel. In D. Büche and N. Hofmann, editors, *COST 526: Automatic Process Optimization in Materials Technology: First Invited Conference*, pages 193–198, Morschach, Switzerland, 2005.
- [3] B. Filipič and E. Laitinen. Model-based tuning of process parameters for steady-state steel casting. *Informatica*, 29(4):491–496, 2005.
- [4] C. M. Fonseca and P. J. Fleming. On the performance assessment and comparison of stochastic multiobjective optimizers. In *Parallel Problem Solving from Nature (PPSN IV)*, volume 1141 of *Lecture Notes in Computer Science*, pages 584–593. Springer, Berlin, 1996.
- [5] K. Price, R. M. Storn, and J. A. Lampinen. *Differential Evolution: A Practical Approach to Global Optimization (Natural Computing Series)*. Springer, Berlin, 2005.
- [6] T. Robič and B. Filipič. DEMO: Differential evolution for multiobjective optimization. In *Proceedings of the Third International Conference on Evolutionary Multi-Criterion Optimization, EMO 2005, Guanajuato, Mexico*, volume 3410 of *Lecture Notes in Computer Science*, pages 520–533. Springer, 2005.
- [7] T. Robič and B. Filipič. Večkriterijsko optimiranje z genetskimi algoritmi in diferencialno evolucijo. Delovno poročilo IJS-DP 9065, Institut “Jožef Stefan”, 2005.
- [8] L. While, P. Hingston, L. Barone, and S. Huband. A faster algorithm for calculating hypervolume. *IEEE Transactions on Evolutionary Computation*, 10(1):29–38, 2006.
- [9] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. da Fonseca. Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Transactions on Evolutionary Computation*, 7(2):117–132, 2003.