UDK 624.057.7+625.08:69.008 Primljeno 1. 12. 2010.

Genetski algoritmi za raspoređivanje rukovatelja građevinskih strojeva

Aleksandar Milajić, Dejan Beljaković, Predrag Petronijević

Ključne riječi

građevinski stroj, rukovatelj, genetski algoritam, raspoređivanje, radni zadaci, učinkovitost

A. Milajić, D. Beljaković, P. Petronijević *Pregledni rad*

Genetski algoritmi za raspoređivanje rukovatelja građevinskih strojeva

U radu je opisana metodologija primjene genetskih algoritama u rješavanju problema optimalnog raspoređivanja rukovatelja građevinskih strojeva na radne zadatke u cilju ostvarenja maksimalnog učinka. Dan je poseban osvrt na utjecaj ljudskog faktora (obučenosti izvršitelja) na učinkovitos sustava. Opisana je metodologija izrade genetskih algoritama i prikazana je njezina primjena na zadani problem uz diskusiju dobivenih rezultata i njihove osetljivosti u funkciji broja varijabli.

Key words

construction machine, operator, genetic algorithm, assignment, work tasks, efficiency

A. Milajić, D. Beljaković, P. Petronijević *Subject review*

Genetic algorithms for assigning tasks to construction machine operators

The authors describe the methodology for using genetic algorithms in solving problems of optimum assignment of tasks to construction machinery operators, aimed at achieving maximum efficiency. A special emphasis is placed on the influence of human factor (operator competence) on the efficiency of the system. The methodology for creating a genetic algorithm is described, and its use in practical problem solving is presented. The achieved results, and their sensitivity to the number of variables, is discussed.

Mots clés

machine de construction, opérateur, algorithme génétique, attribution, tâches de travail, efficacité

A. Milajić, D. Beljaković, P. Petronijević *Ouvrage de syntèse*

Algorithmes génétiques pour attribution des tâches aux opérateurs des machines de construction

Les auteurs décrivent la méthodologie pour la définition des algorithmes génétique visant à résoudre les problèmes d'attribution optimale des tâches aux opérateurs des machines de construction, dans le but d'atteindre une efficacité maximale. L'accent spécial est mis sur l'influence du facteur humain (compétence des opérateurs) sur l'efficacité du système. La méthodologie de développement d'un algorithme génétique est décrite, et son utilisation dans la résolution des problèmes pratiques est présentée. Les résultats obtenus, et leur sensitivité au nombre des variables, est expliquée.

Ключевые слова

строительная машина, оператор, генетический алгоритм, распределение, рабочие задания, эффективность

А. Милайич, Д. Бельякович, П. Петрониевич *Обзорная работа*

Генетический алгоритм для распределения операторов строительных машин

В работе описана методология применения генетических алгоритмов при решении проблемы оптимального распределения рабочих заданий операторам строительных машин в целях достижения максимальной эффективности. Обращено особое внимание на влияние человеческого фактора (обученность исполнителя) на эффективность системы. Описана методология разработки генетических алгоритмов, и показано ее применение для решения заданной проблемы с обсуждением полученных результатов и их чувствительности в зависимости от числа переменных.

Schlüsselworte

Baumaschine, Handhaber, genetischer Algorithmus, Anordnung, Bauaufgaben, Leistungsfähigkeit

A. Milajić, D. Beljaković, P. Petronijević *Übersichtsarbeit*

Genetische Algorithmen für die Anordnung der Handhaber von Baumaschinen

Im Artikel beschreibt man die Methodologie der Anwendung genetischer Algorithmen bei der Lösung des Problems optimaler Anordnung der Hnadhaber von Baumaschinen für Bauaufgaben im Ziel der Realisierung maximaler Leistung. Man gibt einen besonderen Rückblick auf den Einfluss des menschlichen Faktors (Einübung der Ausführer) auf die Leistungsfähigkeit des Systems. Die Methodologie der Ausarbeitung genetischer Algorithmen ist beschrieben und deren Anwendung auf das aufgegebene Problem dargestellt, mit einer Diskussion der erreichten Ergebnisse und deren Empfindsamkeit abhängend von der Zahl der Variablen.

Autori: Mr. sc. **Aleksandar Milajić**, dipl. ing. građ.; mr. sc. **Dejan Beljaković**, dipl. ing. građ., Fakultet za graditeljski menadžment, Beograd; mr. sc. **Predrag Petronijević**, dipl. ing. građ., Građevinski fakultet Univerziteta u Beogradu

1 Uvod

Suvremeni pristup upravljanju projektima, osim visoke razine znanja svih sudionika u realizaciji postavljenog zadatka zahtijeva i najpovoljniji raspored izvršitelja aktivnosti unutar postojećeg sustava [1]. Pod najpovoljnijim se podrazumijeva onaj raspored čijom se realizacijom postiže maksimum funkcije cilja. Stoga je raspoređivanje izvršitelja veoma važan čimbenik za planiranje i uspješnu realizaciju postavljenih zadataka.

Pri rješavanju problema raspoređivanja izvršitelja pojedinih aktivnosti, u konkretnom slučaju pri raspoređivanju rukovatelja građevinskih strojeva, odluke su se donosile i još se najčešće donose na osnovi profesionalnih osjećaja i intuicije, a veoma malo na osnovi egzaktnih metoda i proračuna. Dobro intuitivno osjećanje temelji se na iskustvu i znanju stečenom obrazovanjem, ali sa sobom nosi i manje-više izraženu mogućnost pojave subjektivne pogreške slučajnog karaktera.

Iako konceptualno vrlo jasne, mnoge proračunske metode za rješavanje problema iz domene optimizacije prije su iziskivale dug i složen proračun podložan pogreškama, tako da njihove primjene u praksi dugo nije bilo u mnogim područjima. Međutim, razvoj računalne tehnologije omogućio je izradu i unapređivanje softverskih paketa (npr. WinQSB) koji omogućavaju relativno brzo rješavanje takvih problema, pod uvjetom da je matematički model ispravno formuliran.

Osim gotovih softverskih paketa i programa, stručnjacima su danas na raspolaganju i moduli u okviru programskih paketa *Matlab* i *Excell*, koji korisniku omogućavaju primjenu *state of the art* tehnika programiranja, kao što su genetski algoritmi [2], metode inteligencije rojeva [3], *fuzzy* logika [4] itd.

Ovaj se rad bavi primjenom genetskih algoritama za iznalaženje optimalnog rasporeda rukovatelja građevinskih strojeva na radne zadatke radi postizanja maksimalnog učinka sustava.

2 Učinkovitost sustava

U skladu s razvojem tehnike i tehnologije, u današnje je vrijeme potpuno uobičajeno da je svaki rukovatelj građevinskih strojeva osposobljen da radi na više vrsta i tipova strojeva. Međutim, efikasnost jednog rukovatelja nije ista na svim strojevima za koje je osposobljen. Primjera radi, on može postizati odlične rezultate na dozeru, zadovoljavajuće na pumpi za transport betona, dok bagerom zna upravljati tek na elementarnoj razini.

U praksi se veoma često pojavljuje situacija u kojoj se za izvršavanje nekog zadatka može upotrijebiti više različitih vrsta i tipova strojeva, a na raspolaganju je i odgovarajući broj rukovatelja čija je obučenost na pojedinim strojevima poznata. Potrebno je rukovatelja rasporediti na strojeve tako da se ostvari optimalna učinkovitost sustava, gdje se pod učinkovitosti sustava podrazumijeva stupanj njegove sposobnosti za izvršavanje namijenjenog zadatka, uz angažiranje određenih resursa, precizirane kvalitete, u određenom razdoblju prema određenoj tehnologiji i u određenim uvjetima funkcioniranja [5]. Optimalna je učinkovitost ona koja se može postići samo najpovoljnijim iskorištavanjem raspoloživih resursa i funkcioniranjem komponenata koje se maksimalno prilagođavaju svim uvjetima.

Broj raspoloživih mogućnosti za raspoređivanje rukovatelja na strojeve ovisit će o broju strojeva i broju rukovatelja. Na primjer, ako treba rasporediti dva rukovatelja na dva stroja, na raspolaganju su dvije mogućnosti, jer se svaki rukovatelj može rasporediti na jedan ili drugi stroj. Raspoređivanjem jednog rukovatelja određen je izbor i za drugoga. Povećanjem broja strojeva i rukovatelja do nekog broja *n*, broj raspoloživih mogućnosti $A_{(n)}$ povećava se po zakonu $A_{(n)} = n!$ i jednak je broju permutacija od *n* elemenata. Na primjer, ako treba rasporediti šest rukovatelja na šest strojeva, broj raspoloživih mogućnosti je $A_{(6)} = 6! = 720$, dok bi za slučaj da je broj strojeva i rukovatelja 10, na raspolaganju bilo $A_{(10)} = 10! =$ 3.628.800 mogućnosti.

Ako je broj rukovatelja (n) veći od broja raspoloživih strojeva (k), broj mogućnosti angažiranja rukovatelja na strojevima (A) iznosi:

$$A = \binom{n}{k} = \frac{n!}{k!(n-k)!} \tag{1}$$

Kako je riječ o vrlo velikom broju mogućnosti, očito je da za iznalaženje najpovoljnijega rješenja intuicija i profesionalni osjećaji nisu dovoljni i da se treba osloniti na odgovarajuće proračunske metode procjene.

3 Utjecaj ljudskog faktora

Sposobnost rukovatelja da rukuje nekim strojem veoma je značajna za praktični učinak stroja i pri proračunu praktičnog učinka treba je uzeti u obzir kao koeficijent rukovatelja k_R [6]. Izraz za proračun praktičnog učinka ima kod svih strojeva isti oblik, pri čemu se, dakako, pojavljuju razlike ovisno o tehničkim svojstvima stroja i vrste posla za koji je namijenjen. Tako, na primjer, izraz za proračun praktičnog učinka dozera glasi:

$$U_P = U_T \frac{k_c k_p k_R}{k_r} \tag{2}$$

gdje su:

U_P – praktični učinak dozera

U_T – teorijski učinak dozera

k_c – koeficijent otpora rezanju

k_p – koeficijent punjenja

k_r – koeficijent rastresitosti

k_R – koeficijent rukovatelja

Vrijednost koeficijenta k_R, u izrazima za proračun praktičnog učinka građevinskih strojeva, kreće se od 0 do 1 i izračunava iz izraza:

$$k_R = \frac{O_p}{10} \qquad (3)$$

gdje je O_p ocjena izučenih rukovatelja za rad na pojedinim građevinskim strojevima.

4 Matematička formulacija problema

Da bi se problem optimalnog raspoređivanja rukovatelja na građevinske strojeve mogao riješiti analitički i da bi se između velikog broja mogućih i tehnički ispravnih rješenja pronašlo najpovoljnije, odnosno ono koje će dati najveću učinkovitost sustava, nužno je imati matematički model koji se sastoji od funkcije cilja, čija se ekstremna vrijednost traži (u slučaju učinka sustava to će biti maksimum), i odgovarajućih uvjeta ograničenja kojima će se iz razmatranja eliminirati sva nezadovoljavajuća ili tehnički neizvodljiva (nemoguća) rješenja.

Ako za izvršenje postavljenog zadatka na raspolaganju ima *n* građevinskih strojeva na koje treba rasporediti *m* rukovatelja, pri čemu je sposobnost rukovatelja *i* (i = 1, ...,m) za rad na stroju *j* (j = 1,2,…,n) definirana koeficijentom $k_{R,ij}$, čija je vrijednost poznata, mogu se, prema izrazu (2) proračunati sve vrijednosti U_{ij}, odnosno praktični učinak stroja *j* kada njime upravlja rukovatelj *i* . Od ovih vrijednosti formira se matrica praktičnog učinka *U* (slika 1.), u kojoj svaki redak pripada po jednom rukovatelju, dok su strojevima dodijeljeni stupci.

Strojevi (rukovatelji)	M_1	…	M_j	…	M_m
R_1	U_{11}	…	U_{1j}	…	U_{1m}
…	…	…	…	…	…
R_i	U_{i1}	…	U_{ij}	…	U_{im}
…	…	…	…	…	…
R_n	U_{n1}	…	U_{ni}	…	U_{nm}

Slika 1. Matrica praktičnog učinka U

Cilj raspoređivanja rukovatelja na strojeve jest postizanje najpovoljnije učinkovitosti danog sustava. Stoga se kao kriterij za ocjenu učinkovitosti u ovoj analizi rabio intenzitet učinkovitosti [6], odnosno trenutačni potencijal ili razina učinkovitosti sustava, koji je definiran kao učinak (kapacitet) sustava u odabranoj jedinici vremena.

S obzirom na to da je uobičajeno da se kapacitet građevinskih strojeva izražava u m^3/h, u ovom je slučaju za mjeru intenziteta učinkovitosti usvojena upravo ta jedinica.

Prema navedenom, matematički model problema definiran je odgovarajućom funkcijom cilja i korespondentnim ograničenjima. Funkcija cilja može se definirati kao:

$$\max F_{(x)} = \sum_{i=1}^{m} \sum_{j=1}^{n} x_{ij}\, e_{ij} \qquad (4)$$

gdje su:

$F_{(x)}$ - funkcija cilja (intenzitet učinkovitosti sustava)

e_{ij} - intenzitet učinkovitosti j-tog stroja kad njime rukuje i-ti rukovalac, a proračuna se prema izrazu za za praktični učinak U_p za svaki stroj

x_{ij} - varijabla s vrijednostima 1 ili 0 koja označava je li i-ti rukovatelj raspoređen na j-ti stroj (x_{ij} = 1) ili nije (x_{ij} = 0) .

Navedena definicija varijabli x_{ij} nameće sljedeće uvjete ograničenja:

$$\sum_{j=1}^{n} x_{ij} = 1 \quad (i = 1, 2, ..., m) \qquad (5)$$

$$\sum_{i=1}^{m} x_{ij} = 1 \quad (j = 1, 2, ... n) \qquad (6)$$

uz uvjet:

$$x_{ij} = \begin{cases} 1, \text{ako je i - ti rukovatelj na j - tom stroju} \\ 0, \text{ako i - ti rukovatelj nije na j - tom stroju} \end{cases} \qquad (7)$$

Prvim ograničenjem (5) u ovom matematičkom modelu osigurava se da jedan stroj može biti raspoređen samo jedan rukovatelj, dok drugo ograničenje (6) osigurava da jedan rukovatelj može biti raspoređen samo na jedan stroj.

Kako broj strojeva ne mora biti jednak broju rukovatelja, ovaj zadatak u općenitom slučaju pripada otvorenim problemima. Ako matematički aparat to zahtijeva, svođenje otvorenog problema na zatvoreni može se ostvariti dodavanjem potrebnog broja redova ili stupaca kako bi se dobila kvadratna matrica, s tim što bi u dodatnim redovima i stupcima sve vrijednosti bile jednake nuli.

5 Metode rješavanja

Iako koncepcijski veoma jasan i jednostavan, problem je zbog svoje faktorijelne složenosti veoma težak za rješavanje, čak toliko da pripada grupi NP-teških problema [7]. Kako je funkcija cilja (4) dana kao linearna forma varijabli x_{ij}, upravo kao i uvjeti ograničenja (5, 6), linerano programiranje nameće se kao jedan od mogućih načina određivanja najpovoljnijeg rješenja. S druge strane, oblik funkcije cilja potpuno je isti kao kod funkcije cilja u transportnom problemu linearnog programiranja.

5.1 Linearno programiranje

Zadatak linearnog programiranja sastoji se u iznalaženju optimalne (minimalne ili maksimalne) vrijednosti linearne funkcije cilja, uz zadovoljenje svih zadanih uvjeta ograničenja koji su dani u obliku linearnih jednadžbi i/ili nejednadžbi. Za rješavanje zadatka linearnog programiranja postoji veliki broj algoritama, među kojima je najstariji, i ujedno najpoznatiji, simpleks metoda koja je odavno pronašla primjenu u rješavanju optimizacijskih problema u graditeljstvu [8]. Međutim, ukupan broj uvjeta ograničenja u ovom problemu (5, 6) jednak je *m + n*,odnosno *2m* ako je *m = n*, tako da bi se za veći broj varijabli, a samim tim i za veći broj uvjeta ograničenja, simpleks metoda pokazala neprikladnom zbog vremena potrebnog za unos svih uvjeta i vremena potrebnog za proračun iako bi se kao rezultat sigurno dobilo optimalno rješenje.

5.2 Transportni problem

Kao što je već spomenuto, oblik funkcije cilja (4) u potpunosti odgovara formulaciji funkcije cilja u transportnom problemu linearnog programiranja čija klasična formulacija zadaće glasi:

„Promatra se *m* skladišta koja nude određenu robu u količinama $a_1, a_2, ..., a_m$ i *n* potrošača koji tu robu potražuju u količinama $b_1, b_2, ..., b_n$. Ako su poznate cijene prijevoza od svakog skladišta *i* (*i = 1, ..., m*), do svakog potrošača *j*, (*j = 1, ..., n*), i ako su one dane kao C_{ij}, potrebno je odrediti sve vrijednosti X_{ij} gdje je X_{ij} količina robe koju treba prevesti od skladišta *i* do potrošača *j*, tako da ukupni troškovi transporta budu minimalni“.

Za ovako definirane parametre, funkcija cilja imat će oblik:

$$F = \sum_{i=1}^{m}\sum_{j=1}^{n} C_{ij} X_{ij} \tag{8}$$

Formulacija transportnog problema jasnija je ako se prikaže tablično (tablica 1.).

Ako se pretpostavi da je ukupna ponuda (ukupna količina robe u svim skladištima, odnosno $a_1 + a_2 + ... + a_m$) jednaka ukupnoj potražnji ($b_1 + b_2 + ... + b_n$), iz zadanih uvjeta problema mogu se formulirati *m + n* ograničenja:

$$\sum_{j=1}^{n} x_{ij} = a_i \ (i = 1, 2, ... \ m) \tag{9}$$

$$\sum_{i=1}^{m} x_{ij} = b_j \ (j= 1, 2, ... \ n) \tag{10}$$

Tablica 1. Formulacija transportnog problema

		Potrošači				
		P_1	P_2	…	P_n	Kapaciteti a_i
Skladišta	S_1	C_{11}	C_{12}	…	C_{1n}	a_1
	S_2	C_{21}	C_{22}	…	C_{2n}	a_2
	…	…	…	…	…	…
	S_m	C_{m1}	C_{m2}	…	C_{mn}	a_m
	Potrebe b_j	b_1	b_2	…	b_n	Σ

Iako su funkcija cilja (8) i uvjeti ograničenja (9, 10) istog oblika kao za problem raspoređivanja rukovatelja na strojeve, između njih postoji jedna bitna razlika, a to je činjenica da se u transportnom problemu roba iz jednog skladišta može isporučivati do više potrošača i da svaki potrošač može primati robu iz različitih skladišta, dok kod polaznog problema svaki rukovatelj može posluživati samo jedan stroj i, obratno, svaki stroj može biti posluživan samo od jednog rukovatelja, pri čemu se svi poslovi (ili bar veći dio njih) obavljaju istodobno. Drugim riječima, problem raspoređivanja rukovatelja zapravo je specijalan slučaj transportnog problema u kom je priroda problema gotovo istovjetna problemu trgovačkog putnika [9], s tim što se odstupanje od standardnog oblika ovog problema [10] može uzeti u obzir djelomičnim usvajanjem koncepta rješenja problema 8 kraljica [11], koji zbog faktorijelne složenosti također pripadaju NP-teškim problemima.

5.5 Pretraživanje grubom silom i pohlepni algoritam

Pretraživanje grubom silom (*Brute Force Search*) najjednostavnija je metoda rješavanja NP problema, a svodi se na to da se isprobaju sve permutacije, pri čemu se za svaku od njih vrijednost funkcije cilja uspoređuje s prethodnim rezultatom i pamti se bolji rezultat. Iako se na kraju uvijek dobiva optimalno rješenje, ovaj je postupak zbog svoje jednostavnosti ujedno i najsporiji i gubi smisao već kod malog broja (manje od 10) varijabli [11].

Pohlepni algoritam (*Greedy Algorithm*) je postupak koji se na prvi pogled nameće kao najjednostavniji i najlogičniji, jer se svodi na to da se u matrici praktičnog učinka (slika 1.) pronađe i sačuva maksimalna vrijednost, čime se iz daljeg razmatranja eliminiraju red i stupac u kojima se ona nalazi. Potom se među preostalim raspoloživim vrijednostima nalazi najveća i postupak se ponavlja do kraja. Ovaj je postupak veoma brz, ali vrlo često ne daje najpovoljnije rješenje jer se ne koristi svim podacima (može doći do prerane eliminacije nekih podataka). Osim toga, može se dogoditi da se u jednom trenutku pojave dvije ili više jednakih maksimalnih vrijednosti, tako da bi tada trebalo ispitati sva varijantna rješenja [12].

6 Genetski algoritam

Genetski algoritmi pripadaju skupini pretraživačkih algoritama za određivanje optimalne (minimalne ili maksimalne) vrijednosti funkcije cilja, a inspirirani su biološkim procesima prirodne selekcije i genetike [13]. Od devedesetih godina prošlog stoljeća do danas došlo je do intenzivnog razvoja genetskih algoritama i njihove primjene za rješavanje problema optimizacije u mnogim područjima, uključujući i građevinarstvo [14-16]. Njihova prednost u odnosu na ostale metode optimizacije očituje se, kako su to R. L. Haupt i S. E. Haupt [17] primijetili, u mogućnosti optimizacije i s diskretnim i s parametarski zadanim varijablama, u činjenici da nije nužno poznavati podatke o derivacijama funkcije cilja, u mogućnosti rada s velikim brojem varijabli i u velikom broju točaka u području mogućih rješenja, u mogućnosti primjene čak i za najsloženije funkcije cilja, kao i u mogućnosti određivanja ne samo jednoga lokalnog optimuma, što je slučaj s nekim drugim numeričkim metodama, nego i više optimuma, ako postoje, i globalni optimum (minimum ili maksimum).

Princip rada s genetskim algoritmima bit će u nastavku teksta objašnjen u osnovnim crtama na primjeru rješavanja polazne zadaće, odnosno određivanja optimalnog rasporeda rukovatelja na građevinske strojeve.

6.1 Prikaz rješenja

Osnovna je odrednica svakoga genetskog algoritma kromosom koji se, kao i kod živih bića, sastoji od gena, gdje svaki gen predstavlja jedan parametar, odnosno varijablu x_i ($i = 1, ..., n$). Drugim riječima, svaki kromosom predstavlja jedno moguće rješenje zadaće, odnosno komplet vrijednosti promjenljivih koji zadovoljava polazne uvjete:

$$X = [\text{kromosom}] = [\text{gen}_1, ..., \text{gen}_n] = [x_1, ..., x_n] \qquad (11)$$

Kao i u živom svijetu, svaki kromosom predstavlja jednu jedinku, a skup svih kromosoma čini populaciju, odnosno skup mogućih rješenja problema. Struktura kromosoma ovisi o strukturi samog problema, odnosno o broju podataka koje treba odrediti. U konkretnom primjeru, a prema razmatranju problema N kraljica [18], odabran je permutirani niz, u kome mjesto svakog podatka označava rukovatelja, odnosno vrstu u matrici ulaznih podataka, dok sama vrijednost podatka određuje redni broj stroja (stupac). Na primjer, zapis rješenja u prikazanom primjeru:

$$[3 \quad 2 \quad 6 \quad 4 \quad 1 \quad 8 \quad 7 \quad 5] \qquad (12)$$

znači da se prvi rukovatelj raspoređuje na stroj br. 3, drugi na stroj br. 2, treći na stroj br. 6, itd. Eliminacijom dupliranja u ovako formiranom nizu onemogućava se da se isti rukovatelj pojavi na dva stroja, a predstavljanjem pozicije samo jednim podatkom (rednim brojem stroja) onemogućava se da se na isti stroj rasporede dva rukovatelja.

6.2 Genetski operatori

Na osnovi definiranog kromosoma (jedinke), slučajnim izborom formira se *inicijalna populacija*, odnosno unaprijed odabran dovoljno veliki broj zadovoljavajućih rješenja problema i za svako se određuje vrijednost funkcije kriterija, tj. funkcije cilja. Ova funkcija se u genetskim algoritmima naziva funkcijom podobnosti (*fitness function*). Mjera podobnosti svake jedinke utvrđuje se unaprijed utvrđenim kriterijem. Ako se optimalnom smatra minimalna vrijednost funkcije podobnosti (funkcije cilja), prvo se odredi njezina apsolutno najveća moguća vrijednost (ili se usvoji dovoljno veliki broj), a pogodnost svake jedinke dobiva se kao razlika između najveće moguće vrijednosti i vrijednosti funkcije podobnosti za promatranu jedinku. Ako se traži maksimum funkcije cilja, kao što je slučaj u razmatranom primjeru, jedinke se rangiraju upravo po vrijednosti funkcije cilja, odnosno daje se prednost jedinkama s najvećim vrijednostima funkcije podobnosti.

Procesom prirodne selekcije odbacuju se one jedinke (točke) koje imaju nepodobne vrijednosti funkcije cilja, a preostale se podvrgavaju sparivanju i/ili mutaciji, sve dok se ne dobije nova populacija koja se potom podvrgava istom postupku. Procedura se nastavlja sve dok se ne ispune uvjeti konvergencije, odnosno dok se populacija ne svede na jednu ili nekoliko točaka bliskih optimalnom rješenju, što se vidi na osnovi istih ili veoma bliskih vrijednosti funkcije cilja u nekoliko uzastopnih generacija. [19]

6.2.1 Selekcija

Kao i u prirodi, pod selekcijom se podrazumijeva odabir jedinki koje stvaraju potomstvo, čime se čuvaju i na sljedeću generaciju prenose dobra svojstva (geni), dok se loša svojstva odbacuju. Selekcija se vrši prema podobnosti (*fitness*), tako da jedinke s većom vrijednosti funkcije podobnosti imaju veći izgled da budu odabrane za roditelje novih jedinki.

U praksi je razvijeno više metoda odabira roditelja, ali najčešće se upotrebljavaju: proporcionalni odabir (jedinka s najvećom podobnošću ima najveće izglede da bude odabrana, jedinka s najmanjom ima najmanje izglede, dok se izgledi ostalih jedinki nalaze između ovih vrijednosti), odabir po rangu (jedinke se rangiraju prema podobnosti od najbolje do najgore, a onda se ovisno o rangu određuje koliko će potomaka ostaviti koja jedinka) i turnirski odabir (iz čitave populacije nasumice se bira određen broj jedinki koji se naziva *prozor*, a određen broj najboljih uzima se za roditelje sljedeće generacije i taj se postupak ponavlja dok broj potomaka ne dostigne veličinu populacije).

U skladu s prirodom problema i oblikom kromosoma, u promatranom se primjeru rabila turnirska selekcija [20], s tim što su se u svakom krugu odabira za roditelje sljedeće generacije birale dvije najbolje jedinke u *prozoru*, a ostale su se vraćale natrag u polaznu populaciju. Postupak se ponavlja sve dok se ne popuni sljedeća generacija.

6.2.2 Križanje

U prirodi se križanje genetskog materijala roditelja odvija procesom poznatim kao *crossover*, tokom kojeg dolazi do razmjene gena između očeva i majčina kromosoma, a kao rezultat nastaje kromosom djeteta. Kod genetskih algoritama postupak je potpuno isti, s tim što se ovisno o obliku prikaza rješenja odabire tip križanja koji najbolje odgovara tom izboru. Najčešće se upotrebljava križanje u jednoj točki (bira se isto mjesto u kromosomu obaju roditelja i svim genima koji se u jednom i drugom kromosomu nalaze desno od te točke zamijene se mjesta) i križanje u dvjema točkama (biraju se po dvije točke na istim mjestima u kromosomima roditelja i svim genima između tih točaka zamijene se mjesta). Procesom križanja od dva roditelja dobivaju se dva potomka koji u sebi nose svojstva i jednog i drugog roditelja.

U konkretnoj se zadaći upotrebljavalo križanje u dvjema točkama, uz vođenje računa da se problem ponovljenih vrijednosti u kromosomu rješava na licu mjesta primjenom djelomično usklađenog križanja (*Partially Matched Crossover – PMX*) [19]. Na primjer, ako su odabrani roditelji i točke prekida prikazani kao:

$$[3 \quad 2 \quad 6 \mid 4 \quad 1 \quad 8 \mid 7 \quad 5]$$

$$[5 \quad 8 \quad 3 \mid 2 \quad 4 \quad 6 \mid 1 \quad 7]$$

razmjenom gena između točaka prekida dobivaju se dvije nove jedinke:

$$[3 \quad 2 \quad 6 \mid 2 \quad 4 \quad 6 \mid 7 \quad 5]$$

$$[5 \quad 8 \quad 3 \mid 4 \quad 1 \quad 8 \mid 1 \quad 7]$$

Očito je da je kod obadvije jedinke došlo do ponavljanja gena u području izvan razdjelnih točaka (2 i 6 kod prvog potomka, 1 i 8 kod drugog). Ovaj se problem rješava tako da se konfliktne vrijednosti na starijim pozicijama (izvan razdjelnih točaka) mijenjaju vrijednostima koje su stajale na novim pozicijama prije križanja. Tako će u promatranom primjeru kod prvog potomka gen 2 biti prvo zamijenjen genom 4 a potom genom 1 (dvostruka promjena 2–4–1) [19], dok će gen 6 biti zamijenjen genom 8, a kod drugog potomka će gen 1 biti prvo zamijenjen genom 4 a potom genom 2, dok će gen 8 biti zamijenjen genom 6, čime se dobivaju dva „ispravna“ potomka, odnosno zadovoljava se polazni uvjet neponavljanja istih vrijednosti:

$$[3 \quad 1 \quad 8 \mid 2 \quad 4 \quad 6 \mid 7 \quad 5]$$

$$[5 \quad 6 \quad 3 \mid 4 \quad 1 \quad 8 \mid 2 \quad 7]$$

6.2.3 Mutacija

Mutacija se u prirodnim procesima definira kao slučajna promjena gena. U genetskim algoritmima, međutim, mutacija se rabi kao genetski operator kojim se osigurava genetska raznolikost nove generacije u odnosu na prethodnu. Ovim se sprječava da rješenja postanu previše slična, čime bi se evolucija (konvergencija prema optimalnom rješenju) usporila ili čak potpuno zaustavila. Bez mutacija proces bi najčešće konvergirao k lokalnom minimumu umjesto prema globalnom.

U promatranom primjeru odabran je najjednostavniji operator mutacije koji samo zamjenjuje mjesta dvjema slučajno odabranim vrijednostima unutar jedne n-tice, a time se čuva uvjet o neponavljanju vrijednosti, tako da, na primjer, od kromosoma:

$$[3 \quad 1 \quad 8 \mid 2 \quad 4 \quad 6 \mid 7 \quad 5]$$

nastaje kromosom:

$$[3 \quad 1 \quad 7 \mid 2 \quad 4 \quad 6 \mid 8 \quad 5]$$

6.2.4 Kriterij završetka

Proces stvaranja novih generacija ponavlja se sve dok se ne zadovolji unaprijed utvrđeni kriterij završetka evolucije. To može biti trenutak kad se pronađe rješenje koje zadovoljava minimalni kriterij pogodnosti, trenutak kad se dostigne unaprijed odabrani broj generacija ili kad se evolucija praktično završi, a to se očituje u tome što nove iteracije ne daju bolje rezultate.

Tokom rješavanja zadanog problema istraživala su se dva različita slučaja, jednom s unaprijed određenim brojem generacija, a drugi put do dostizanja optimuma, odnosno do završetka konvergencije.

7 Rezultati i osjetljivost rješenja

Autori ovog rada primijenili su opisane korake i postupke u pisanju odgovarajućeg programa u okviru programskog paketa *Matlab*. Analizirajući različite kombinacije ulaznih podataka, odnosno različite brojeve rukovatelja i strojeva i istražujući utjecaj ponavljanja vrijednosti učinaka na brzinu konvergencije, kao i osjetljivost rješenja, ustvrdili su da za probleme manjeg opsega (do 8 rukovatelja), algoritam uvijek nalazi optimum, i to nakon svega tri ili četiri generacije, dok se za složenije slučajeve (15, 20 i 30 varijabli) broj iteracija do nalaženja optimuma razumno povećava (12, 16, 19). U složenijim primjerima najveća promjena funkcije cilja dogodila se u prvih 4–7 iteracija, da bi se postupak potom sve više usporavao (ponovljene vrijednosti u po 3–4 uzastopne generacije) sve do dostizanja konačne (optimalne) vrijednosti koja je ujedno i rješenje zadatka.

8 Zaključak

Genetski su algoritmi relativno brzo prešli put od smjele ideje, zasnovane na biološkom procesu selekcije i reprodukcije živih organizama, do široke primjene u rješavanju problema optimizacije i to u svim područjima uključujući i građevinarstvo. Njihova golema prednost u od nosu na druge numeričke i aproksimativne metode optimizacije očituje se u tome što kod genetskih algoritama funkcija cilja ne samo da ne mora biti neprekidna, nego može biti i veoma složena i s velikim brojem varijabli, kao i uvjeti ograničenja. Osim toga, dovoljno dugim procesom evolucije i nekoliko puta ponovljenim proračunom s istim ulaznim podacima efikasno se rješava problem lokalnih ekstrema funkcije cilja, ako postoje. Primjena genetskog algoritma pokazala se vrlo uspješnom i opravdanom u slučaju rješavanja problema raspoređivanja rukovatelja na građevinske strojeve budući da je ispisivanjem relativno jednostavnog programa u jeziku *Matlab* ovaj NP-težak problem riješen u iznimno kratkom vremenu i sa zadovoljavajućim rezultatima. Dalji rad na istraživanju mogućnosti rješavanja ovog problema mogao bi obuhvatiti i uključivanje vremena trajanja pojedinih aktivnosti, kao i količine radova, čime bi se došlo do još povoljnijih rezultata i omogućilo podizanje menadžmenta ljudskih resursa u građevinarstvu na višu razinu u skladu sa zahtjevima održivog razvoja i održive gradnje.

LITERATURA

[1] Štromar, Ž.: *Organizacijski model upravljanja gradnjom*, Građevinar 61 (2009) 6, 557-562

[2] Camp, C.V.; Pezeshk, S., Hansson, H.: *Flexural Design of Reinforced Concrete Frames using a Genetic Algorithm*, Journal of Structural Engineering, 129 (2003) 1, 105-115

[3] Praščević, Ž.; Praščević, N.: *Optimizacija pomoću članova rojeva*, Izgradnja 62 (2008), 8-9, 339-345

[4] Soh, C.K.; Yang, J.: *Fuzzy Controled Genetic Algorithm Search for Shape Optimization*, Journal of Computing in Civil Engineering, 10 (1996) 2, 143-150

[5] Pinter, U.; Lončarić, R.: *Značaj studija rada u građevinarstvu*, Građevinar 58 (2006) 10, 807-812

[6] Kurij, K.; Kostić, Lj.; Hadžić, D.: *Programiranje u istraživanju pouzdanosti građevinskih mašina*, Naučna knjiga, Beograd, 1987.

[7] Schwefel, H.P.: *Numerical Optimization of Computer Models*, John Willey & Sons, New York, 1981.

[8] Stevovic, S.; Milovanovic, Z.; Milajic, A.: *New Methodological Approach in Techno-economic and Environmental Optimization of Sustainable Energy Production*, Thermal Science, 14 (2010) 3, 809-819

[9] Applegate, D. L.; Bixby, R. E.; Chavatal, V.; Cook, W. J.: *The Traveling Salesman Problem: A Computational Study*, Princeton University Press, 2006.

[10] Vickers, D.; Butavicius, M.; Lee, M.; Medvedev, A.: *Human Performance on Visually Presented Traveling Salesman Problems*, Psychological Research, 65 (2001) 1, 34–45

[11] Bell, J.; Stevens, B.: *A Survey of Known Results and Research Areas for N-Queens*, Discrete Mathematics, 309 (2009) 1, 1-31.

[12] Cormen, T.H.; Leisersin, C.E.; Rivest, R.L.; Stein, C.: 35.2: *The traveling-salesman problem,* Introduction to Algorithms, 2nd edn, MIT Press and McGraw-Hill, 2001.

[13] Marin, G.: Genetski algoritam, I dio, *www.zemris.fer.hr/~golub/ga/ga_skripta1.pdf*, (1997)

[14] Praščević, Ž.: *Binarni genetski algoritmi za optimizaciju sustava*, Izgradnja 58 (2004) 3-4, 55-68

[15] Rajeev, S.; Krishnamoorthy, C.S.: *Discrete Optimization of Structures Using Genethic Algorithms,* Journ. Of Structural Engineering, 118 (1992) 5, 1233-1250.

[16] Chen, S-Y.; Rajan, S.D.: *A Robust Genetic Algorithm for Structural Optimization*, Structural Engineering and Mechanics Journal, 10 (2000)4, 313-336

[17] Haupt R.E.; Haupt, S.E.: *Practical Genetic Algorithms*, John Willey & Sons, New York, 1998

[18] Goos J.: *Using Genetic Programing To Solve The N-Queens Problem*, (2005), Roskilde Universitetscenter, http://akira.ruc.dk/~henning/KIIS05/KIIS05AssignFromStudents/JGoos/assignment.pdf

[19] Goldberg, D. E. (1989), *Genetic Algorithms in Search, Optimization & Machine Learning*, New York: Addison-Wesley

[20] Yang, J.; Soh, C.K.: *Structural Optimization by Genetic Algorithms with Tournament Selection*, ASCE Journal of Structural Engineering, 11 (1997) 3, 195-200