

## МОДИФИКАЦИЈА PSO АЛГОРИТМА: ПРОМЕНЉИВА ВЕЛИЧИНА ПОПУЛАЦИЈЕ, ХИБРИДИЗАЦИЈА СА НЕЛДЕР-МИД МЕТОДОМ И РЕШЕЊЕ ЈЕДНОГ ПЕРМУТАЦИОНОГ ПРОБЛЕМА

## MODIFICATION OF THE PSO ALGORITHM: VARIABLE POPULATION SIZE, HYBRIDIZATION WITH THE NELDER-MEAD METHOD, AND SOLVING A PERMUTATION PROBLEM

Стефан Топалов, Факултет техничких наука, Нови Сад

Област – ЕЛЕКТРОТЕХНИКА И РАЧУНАРСТВО

**Кратак садржај** – Тема овог рада је решавање оптимизационих проблема применом алгоритама насталих модификовањем PSO алгоритама, са циљем уштеде процесорских ресурса. Први алгоритам који ће бити представљен је комбинација PSO алгоритама са променљивом величином популације, предложеним у [1], са Нелдер-Мид алгоритмом. У другом делу рада, биће приказан један начин решавања проблема трговачког путника (енгл. *Travelling Salesman Problem*). Ефикасност алгоритама је нумерички потврђена.

**Кључне речи:** PSO алгоритам, Нелдер-Мид алгоритам, проблем трговачког путника

**Abstract** - This paper discusses solving optimization problems using modified versions of the PSO algorithm, aiming to reduce computational resource consumption. The first approach presented combines the PSO algorithm, featuring a variable population size as proposed in [1], with the Nelder-Mead method. The second part of the paper demonstrates a solution to the Travelling Salesman Problem. The effectiveness of these algorithms is validated through numerical experiments.

**Keywords:** PSO algorithm, Nelder-Mead algorithm, Travelling Salesman Problem

### 1. УВОД

У овом раду биће представљена два алгорита за решавање типа проблема са којима се свакодневно сусрећемо, а то су оптимизациони проблеми. Први алгоритам решавања оптимизационих проблема који ће бити представљен, може се назвати „хибридни” јер комбинује модификовани PSO (енгл. *Particle Swarm Optimization*) са Нелдер-Мид (енгл. *Nelder-Mead*) алгоритмом. Преласком са модификованог PSO на Нелдер-Мид, постиже се ефикаснија претрага простора у блиској околини

оптималног решења, у смислу тачности, што је поткрепљено нумеричким резултатима.

Други алгоритам намењен је решавању проблема трговачког путника (енгл. *Travelling Salesman Problem*). У циљу проналаска најкраће путање за обилазак свих потребних градова, уместо генетског алгорита који се најчешће користи за решавање овог проблема, употребљен је модификовани PSO алгоритам и упоређене перформансе са генетским. Циљ имплементираних алгоритама је постизање задовољавајућих резултата претраге, уз уштеду меморијских и временских ресурса.

### 2. PSO АЛГОРИТАМ

Алгоритам оптимизације ројем честица осликава интеракцију и сарадњу међу јединкама унутар одређене популације, најчешће у потрази за храном, и тако решава оптимизационе проблеме. Утемељен је 1995. године од стране истраживача Џејмса Кенеџија и Расела Еберхарта [2]. У контексту алгоритама, изједначени су појмови честице и јединке, као и роја и популације. Свака јединка представља потенцијално решење проблема, а цео систем се базира на концепту колективног учења, где се позиције и брзине сваке честице ажурирају у зависности од њихових најбољих постигнућа и постигнућа других честица у роју. Ова метода је посебно корисна за решавање комплексних оптимизационих проблема са великом просторном променљивошћу [3].

### 3. PSO АЛГОРИТАМ СА ПРОМЕНЉИВОМ ВЕЛИЧИНОМ ПОПУЛАЦИЈЕ

PSO алгоритам захтева рачунање нових позиција и вредности критеријумске функције у свакој итерацији, што може бити процесорски захтевно у случају комплексних функција. Модификације из [1] смањују број позива те функције, оптимизујући алгоритам смањењем броја честица током извршавања. Ове модификације омогућавају значајну уштеду ресурса без битног утицаја на квалитет решења, чиме се побољшава ефикасност PSO алгоритама.

#### 3.1. Прва модификација

У првој модификацији алгоритама, критеријум за уклањање честица из популације представља промена вредности најбоље сопствене вредности кри-

### НАПОМЕНА:

Овај рад проистекао је из мастер рада чији је ментор била др Мирна Радовић, ванр. проф.

теријума оптималности. Уколико се најбоља сопствена вредност критеријума оптималности неке честице није променила унапред задати број итерација, а да при томе то није вредност која је уједно и најбоља вредност критеријума оптималности на нивоу целог роја, може се сматрати да честица не претражује простор довољно ефикасно и да нема велики утицај на проналазак оптималног решења, или да је запала у локални оптимум, па се она одстрањује из популације.

### 3.2. Друга модификација

У другој модификацији алгоритма, критеријум за елиминацију честица је међусобно еуклидско растојање. Како алгоритам напредује, честице се групишу око глобалног оптимума, често се налазећи на истој позицији, што доводи до дуплирања рачунања критеријумске функције. Уколико је растојање између честица мање од одређеног прага, честица са слабијом вредношћу критеријума се уклања, смањујући непотребно трошење ресурса без губитка на ефикасности у проналажењу оптимума.

## 4. ХИБРИДИЗАЦИЈА PSO АЛГОРИТМА

### 4.1. Нелдер-Мид алгоритам

Нелдер-Мид алгоритам (енгл. *Nelder-Mead*) је једноставна метода оптимизације, првобитно намењена за минимизацију функција више променљивих. Предложен је 1965. године од стране научника Џона Нелдера и Роџера Мида [4]. Алгоритам подразумева иницијализацију почетне геометријске структуре, познате као симплекс, са  $n+1$  тачака у  $n$ -димензионалном простору. Након тога, симплекс итеративно пролази кроз низ трансформација које би га помериле ка оптималном решењу. Трансформације подразумевају замену темена са најлошијом вредношћу критеријума неком другом, израчунатом тачком и оне могу бити рефлексија (енгл. *reflection*), експанзија (енгл. *expansion*) и контракција (енгл. *contraction*) или, ако ове трансформације не успеју, скупљање (енгл. *shrinkage*) приликом ког се добија  $n$  нових темена.

### 4.2. Опис алгоритма

PSO алгоритам је веома значајан за глобалну претрагу, али успорава када се честице нађу близу оптимума. С друге стране, Нелдер-Мид алгоритам је прецизан за локалну оптимизацију. Комбинацијом ова два метода, PSO се прво користи за широку претрагу, а затим, након одређеног броја итерација или када број честица опадне на број потребан за формирање симплекса, Нелдер-Мид преузима за фино подешавање решења. Ова стратегија балансира глобалну и локалну оптимизацију, побољшавајући брзину конвергенције и проналазак оптимума.

### 4.3. Нумерички резултати

У циљу истраживања перформанси претходно описаног алгоритма, извршено је тестирање на две стандардне тест функције за оптимизационе проблеме, са две и пет димензија, и резултати су упо-

ређени са стандардним PSO алгоритмом и PSO алгоритмом са променљивом величином популације (модификовани PSO). За обе тест функције, алгоритми су извршени по сто пута и приказани су упросечени резултати. Приликом анализе, број честица иницијалне популације је износио 100, а број итерација је зависио од димензионалности проблема и износио је  $100n$ .

#### 4.3.1. Аклијева функција

Ова тест функција дефинисана је формулом

$$f(x) = 20 + e - 20 \exp\left(-\frac{1}{5} \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}\right) - \exp\left(-\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)\right), \quad (1)$$

где је  $n$  димензионалност простора. Њен глобални минимум је  $f(x^*) = 0$ . Иницијална популација дефинисана је на простору претраге  $-15 \leq x_i \leq 15$ ,  $i = 1, 2, \dots, n$ .

У табели 1 приказани су резултати тестирања алгоритма на Аклијевој функцији са 2 димензије. Ако је са  $g_{\text{best}}$  означено пронађено оптимално решење и са  $m$  број позива функције за рачунање вредности критеријума оптималности, можемо закључити да уведени алгоритам комбинује предности класичног PSO алгоритма и PSO алгоритма са променљивом величином популације, те као резултат имамо подједнако добро решење проблема као и код класичног PSO, уз много мањи број позива функције за рачунање вредности критеријума, што је и био циљ увођења алгоритма описаног у раду.

Табела 1: Анализа Аклијеве функције за две димензије

| Акклијева функција | $g_{\text{best}}$       | $m$     |
|--------------------|-------------------------|---------|
| PSO                | $1.6520 \cdot 10^{-15}$ | 20000   |
| Модификовани PSO   | $3.5302 \cdot 10^{-6}$  | 4147.25 |
| PSO + Нелдер-Мид   | $1.9717 \cdot 10^{-15}$ | 4184.4  |

Из табеле 2 видимо да комбинација PSO и Нелдер-Мид алгоритма у потпуности оправдава разлог имплементације. Скоро у потпуности је очуван квалитет пронађеног решења уз знатно мањи број позива критеријумске функције.

Табела 2: Анализа Аклијеве функције за пет димензија

| Акклијева функција | $g_{\text{best}}$       | $m$     |
|--------------------|-------------------------|---------|
| PSO                | $3.6415 \cdot 10^{-15}$ | 50000   |
| Модификовани PSO   | $8.2464 \cdot 10^{-6}$  | 8311.09 |
| PSO + Нелдер-Мид   | $8.2796 \cdot 10^{-14}$ | 6904.11 |

#### 4.3.2. Гриванкова функција

Гриванкова функција дата је формулом

$$f(x) = \sum_{i=1}^n \frac{x_i^2}{4000} - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1, \quad (2)$$

где је  $n$  димензионалност простора. Њен глобални минимум је  $f(x^*) = 0$ , а поред глобалног, поседује

и велики број равномерно распоређених локалних минимума. Иницијална популација је дефинисана на простору претраге:

$$-600 \leq x_i \leq 600, i = 1, 2, \dots, n.$$

Приликом тестирања на Гриванковој функцији, показало се да је за случај две и пет димензија, алгоритам потпуно надмашио и класични *PSO* и *PSO* са променљивом величином популације. У табели 3 може се видети да је алгоритам у сваком покретању пронашао тачно оптимално решење, уз значајно мањи број позива критеријумске функције у односу на класични *PSO*, док нам табела 4 показује да је решење скоро не приметно нарушено, уз такође значајно мањи број позива критеријумске функције.

Табела 3: Анализа Гриванкове функције за две димензије

| Гриванкова функција | $g_{best}$             | m       |
|---------------------|------------------------|---------|
| PSO                 | 0                      | 20000   |
| Модификовани PSO    | $2.9796 \cdot 10^{-5}$ | 2981.57 |
| PSO + Нелдер-Мид    | 0                      | 3116.28 |

Табела 4: Анализа Гриванкове функције за пет димензија

| Гриванкова функција | $g_{best}$              | m       |
|---------------------|-------------------------|---------|
| PSO                 | $1.1102 \cdot 10^{-18}$ | 50000   |
| Модификовани PSO    | $3.0651 \cdot 10^{-7}$  | 7946.73 |
| PSO + Нелдер-Мид    | $1.7763 \cdot 10^{-17}$ | 7379.84 |

## 5. РЕШАВАЊЕ ПРОБЛЕМА ТРГОВАЧКОГ ПУТНИКА

Проблем трговачког путника (енгл. *Travelling Salesman Problem - TSP*) представља један од најзначајнијих комбинаторних оптимизационих проблема. Формулисан средином 20. века [5], одговара на питање којим редоследом трговац треба да обиђе одређене градове, тако да сваки град посети тачно једном и да се врати у град из ког је и кренуо. Циљ је пронаћи најкраћу могућу руту која повезује све градове, минимизујући дужину пређеног пута.

Да би се смањио утрошак процесорских ресурса, меморије и времена, проблем пермутација ће се решавати *PSO* алгоритмом са променљивом величином популације. Сваку честицу у роју представљаће једна могућа пермутација градова. То се постиже трансформацијом координата честице у индексе сортираних координата, где сваку координату замењује њен индекс у сортираном низу од најмање до највеће вредности. Тако трансформисана позиција се затим користи за израчунавање критеријума оптималности, односно укупног пређеног пута на основу матрице растојања између градова.

Прва модификација примењена је на истоветан начин као што је раније објашњено.

У другој модификацији, адаптиран је начин мерења растојања између честица. Уместо еуклидског растојања, сада се користи разлика у редоследу обиласка градова. За сваке две честице, сабирају се апсолутне вредности разлике бројева на одговарајућим индексима пермутације градова. Ако је ово растојање веће од дефинисаног прага, честица са лошијом вредношћу критеријума оптималности се уклања из популације. Такође, ако су вредности критеријума идентичне, једна од честица се уклања из популације.

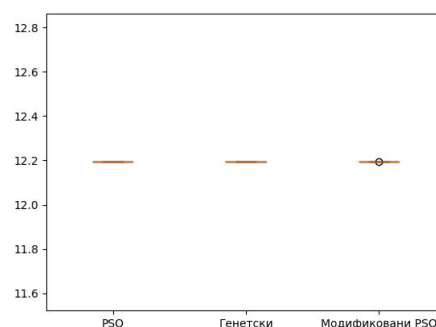
### 5.1. Нумерички резултати

Ради оцене ефикасности предложеног алгоритма, извршено је тестирање на проблемима са седам и десет градова. Резултати су упоређени са класичним *PSO* и генетским алгоритмом. С обзиром на присутност стохастике, сваки алгоритам је извршен сто пута, а затим је спроведена анализа резултата. Оцену перформанси представљају дијаграми који приказују распоред пронађених оптимума.

Приликом тестирања алгоритма на проблему трговачког путника са седам градова, величина иницијалне популације је износила 300 честица, а број итерација 100. У табели 5 налазе се дистанце између градова. У овом случају, модификовани *PSO* даје једно одступање, видљиво на слици 1, које се на основу анализе пронађених оптимума у сваком извршавању, може приписати грешци заокруживања. Узимајући у обзир време извршавања од 74.86 секунди за класичан, 18.51 секунд за модификовани *PSO* и 153.97 секунди за генетски алгоритам, може се закључити да је експеримент за овај случај успешан.

Табела 5: Матрица растојања за проблем седам градова.

|   | 1    | 2    | 3    | 4    | 5    | 6    | 7    |
|---|------|------|------|------|------|------|------|
| 1 | 0.00 | 0.71 | 7.76 | 5.10 | 1.71 | 4.50 | 4.70 |
| 2 | 0.71 | 0.00 | 7.29 | 3.38 | 3.75 | 5.54 | 6.23 |
| 3 | 7.76 | 7.29 | 0.00 | 7.80 | 2.24 | 3.78 | 1.34 |
| 4 | 5.10 | 3.38 | 7.80 | 0.00 | 2.01 | 0.63 | 6.11 |
| 5 | 1.71 | 3.75 | 2.24 | 2.01 | 0.00 | 5.09 | 0.64 |
| 6 | 4.50 | 5.54 | 3.78 | 0.63 | 5.09 | 0.00 | 4.76 |
| 7 | 4.70 | 6.23 | 1.34 | 6.11 | 0.64 | 4.76 | 0.00 |



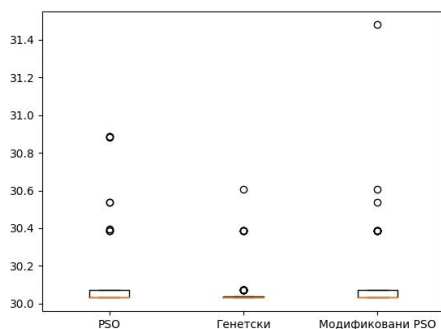
Слика 1: Приказ пронађених оптимума за проблем седам градова.

Са повећањем броја градова број могућих пермутација значајно расте. Из тог разлога, потребан је и већи број честица и итерација за решавање проблема десет градова, те је у овом тестирању коришћено 2000 честица и 500 итерација.

Дистанце између градова приказане су у табели 6. Очекивано, долази до појаве више одступања, што је видљиво на слици 2. На основу ових података, може се утврдити да је оптимизација генетским алгоритмом у овом случају оптимална. Али, како је један од разлога имплементације предложеног алгоритма и уштеда времена приликом извршавања, а класичан *PSO* се, у овом случају, извршавао приближно 36 минута, модификовани 5 минута, а генетски близу 8 сати, може се констатовати да постоји пуно простора за његово значајно унапређење повећањем броја честица и итерација, и постизање бољих резултата у односу на генетски алгоритам.

Табела 6: Матрица растојања за проблем десет градова.

|    | 1    | 2    | 3    | 4    | 5    | 6    | 7    | 8    | 9    | 10   |
|----|------|------|------|------|------|------|------|------|------|------|
| 1  | 0.00 | 1.44 | 2.49 | 5.97 | 2.24 | 4.61 | 9.09 | 4.50 | 3.43 | 3.37 |
| 2  | 1.44 | 0.00 | 1.44 | 7.96 | 2.46 | 6.45 | 7.18 | 4.99 | 4.91 | 4.34 |
| 3  | 2.49 | 1.44 | 0.00 | 3.63 | 4.97 | 3.44 | 3.15 | 6.87 | 2.68 | 7.54 |
| 4  | 5.97 | 7.96 | 3.63 | 0.00 | 6.43 | 5.44 | 3.19 | 4.45 | 2.82 | 4.61 |
| 5  | 2.24 | 2.46 | 4.97 | 6.43 | 0.00 | 5.13 | 5.79 | 4.47 | 2.56 | 7.05 |
| 6  | 4.61 | 6.45 | 3.44 | 5.44 | 5.13 | 0.00 | 4.86 | 6.52 | 5.24 | 5.92 |
| 7  | 9.09 | 7.18 | 3.15 | 3.19 | 5.79 | 4.86 | 0.00 | 2.30 | 1.87 | 4.58 |
| 8  | 4.50 | 4.99 | 6.87 | 4.45 | 4.47 | 6.52 | 2.30 | 0.00 | 3.54 | 4.70 |
| 9  | 3.43 | 4.91 | 2.68 | 2.82 | 2.56 | 5.24 | 1.87 | 3.54 | 0.00 | 5.52 |
| 10 | 3.37 | 4.34 | 7.54 | 4.61 | 7.05 | 5.92 | 4.58 | 4.70 | 5.52 | 0.00 |



Слика 2: Приказ пронађених оптимума за проблем десет градова.

## 6. ЗАКЉУЧАК

У овом раду предложена су могућа унапређења *PSO* алгоритма за решавање оптимизационих проблема. У уводним поглављима описане су теоријске основе самог алгоритма, а затим објашњени детаљни имплементације модификација у [1], чији је резултат *PSO* алгоритам са променљивом величином популације, коришћен као основа за решавање оптимизационих проблема у овом раду. Извршена су

два експеримента и закључци поткрепљени нумеричким резултатима.

Комбиновање *PSO* алгоритма са променљивом величином популације и Нелдер-Мид алгоритма, са циљем мањег утrophка ресурса у односу на класични *PSO* и постизања бољих резултата у односу на модификовани *PSO*, показало се као веома ефикасно и оправдало је разлог имплементације.

Други експеримент, решавање проблема трговачког путника применом *PSO* алгоритма са променљивом величином популације, такође се показао као успешан, уз могућности напредовања за проблем већег броја градова.

У случају да је неко од добијених решења незадовољавајуће, и потребно је прецизније коначно решење, то се може постићи повећањем броја честица иницијалне популације, повећањем броја итерација или корекцијом осталих параметара коришћених приликом тестирања алгоритама.

Могућности иновација на пољу оптимизационих алгоритама остају неограничене.

## 7. Литература

- [1] Стефан Топалов, „Модификовање *PSO* алгоритма променљивом величином популације”, Факултет техничких наука, Универзитет у Новом Саду, 2023.
- [2] J. Kennedy and R. Eberhart, „Particle swarm optimization”, Proceedings of ICNN'95 - International Conference on Neural Networks, Perth, WA, Australia, 1995, pp. 1942-1948 vol.4, doi: 10.1109/ICNN.1995.488968.
- [3] Жељко Кановић, Зоран Јеличић, Милан Рапаић „Еволутивни оптимизациони алгоритми у инжењерској пракси”, Факултет техничких наука, Нови Сад, 2017.
- [4] Nelder, John A. and Roger Mead. “A Simplex Method for Function Minimization.” Comput. J. 7 (1965): 308-313.
- [5] Dantzig, G. B, Fulkerson, R, and Johnson, S. M. (1954). „Solution of a large-scale traveling-salesman problem”, Operations Research, 2(4), 393-410.

## Кратка биографија:



**Стефан Топалов** рођен је у Новом Саду 2000. године. Мастер рад на Факултету техничких наука из области Електротехнике и рачунарства одбра- нио је 2024. године.

Контакт: topalov.stefan.ns@gmail.com