Generalized Net Model of Coyote Optimization Algorithm

Olympia Roeva[1,2,*], Dafina Zoteva[3], Peter Vassilev[1]

[1]*Institute of Biophysics and Biomedical Engineering*
Bulgarian Academy of Sciences
Acad. G. Bonchev Str., Bl. 105, Sofia 1113, Bulgaria
E-mails: olympia@biomed.bas.bg, peter.vassilev@gmail.com

[2]*Department of Mehatronic Bio/Technological Systems*
Institute of Robotics, Bulgarian Academy of Sciences
1113 Sofia, Bulgaria

[3]*Faculty of Mathematics and Informatics*
Sofia University "St. Kliment Ohridski"
5 James Bourchier Blvd., Sofia 1164, Bulgaria
E-mail: dafy.zoteva@gmail.com

[*]*Corresponding author*

Received: July 06, 2021 **Accepted: February 02, 2022**

Published: December 31, 2022

***Abstract**: In the presented paper, the functioning of the coyote optimization algorithm (COA) is described using the apparatus of generalized nets (GNs). The COA is a population-based metaheuristic for optimization inspired by the Canis latrans species. Based on a Universal GN-model of population-based metaheuristics, a GN-model of COA is constructed by setting different characteristic functions of the GN-tokens. The presented GN-model successfully describes the considered metaheuristic algorithm, conducting basic steps and performing an optimal search.*

***Keywords:** Coyote optimization algorithm, Generalized net, Metaheuristic, Population-based algorithms.*

Introduction

One of the contemporary fields of artificial intelligence is the field of metaheuristic algorithms – a scientific method for problem-solving that extends the idea of heuristic algorithms, where "meta" denotes "beyond" or "on a higher level" [7]. Metaheuristic algorithms can be divided into three main groups [6, 14]: (i) evolutionary algorithms (genetic algorithms, genetic programming, differential evolution, evolutionary strategies, evolutionary programming and harmony search); (ii) population-based algorithms (artificial bee colony, particle swarm optimization, ant colony optimization, antlion optimizer, coyote optimization algorithm, etc.); (iii) trajectory-based algorithms (tabu search, simulated annealing, hill climbing).

The coyote optimization algorithm (COA) is a new population-based metaheuristic algorithm for optimization inspired by the *Canis latrans* species that dwells mainly in North America. The algorithm [10] considers the social organization of the coyotes and their adaptation to the environment. It contributes with a different algorithmic structure from other metaheuristics known in the literature. It also provides new mechanisms for balancing exploration and exploitation in the optimization process [10].

The COA is used to solve various problems such as parameters extraction of three-diode photovoltaic models [11], optimization of the location and sizing of solar photovoltaic distribution generation units [9], optimal parameters estimation of proton exchange membrane fuel cell model [16], determination of proportional integral controller gains for enhancing the performance of solar PV water-pumping system [1], etc. The presented results show that the COA is an efficient and robust algorithm.

The paper aims to describe the functioning and results of the COA using the apparatus of generalized nets (GNs) [2]. A detailed description of any algorithm in terms of GNs can lead to a better understanding of the inner workings of the algorithm and consequently to an improvement of its performance [3].

GNs have been proven to be a suitable and efficient tool for modeling various systems [8, 12, 15] and for describing the essence of various optimization methods [4]. So far, GNs have been applied to model metaheuristic algorithms, such as ant colony optimization [5], cuckoo search algorithm [13], etc.

The paper is organized as follows: background notes on COA are presented in Section 2. The developed GN-model of COA is discussed in Section 3. Concluding remarks and further work directions are given in Section 4.

Coyote optimization algorithm

The COA was introduced in [10]. It is a metaheuristic procedure loosely based on the behaviour of coyotes (*Canis latrans*) in nature. Notable differences are the lack of a second alpha coyote and the fixed number of coyotes in a pack.

In what follows, a summary of COA is provided. The population is divided into N_p packs each consisting of $N_c < 15$ coyote individuals. Each individual is a possible decision variable (i.e., has the same size as the search space D).

Each coyote is initialized as

$$soc_{c,j}^{p,0} = LB_j + r_j\left(UB_j - LB_j\right), \tag{1}$$

where j runs in {1, 2, …, D}, LB is the lower bound and UB the upper bound for the j-th component of the decision variable and r_j is a random number uniformly distributed in [0, 1]. The *adaptation* of each coyote can be calculated next as:

$$fit_c^{p,0} = f\left(soc_c^{p,0}\right), \tag{2}$$

where f is the objective function of the problem being solved.

Until a stopping criterion is reached, sequentially do the following.

- For each pack:
  1) Find the *alpha-coyote* (best solution) $min_c \ f\left(soc_c^{p,t}\right)$.
  2) Find the social tendency of the pack *cult* (the median value of the solutions of the pack).

3) For each coyote update the possible new candidate's social value (while still retaining the old) as:

$$new\_soc_c^{p,t} = soc_c^{p,t} + r_1\delta_1 + r_2\delta_2, \quad (4)$$

where r_1 and r_2 are weights of influence of the alpha coyote and the pack, respectively.

They are initialized as random numbers in the unit interval.

δ_1 = *alpha* – *soc* (random coyote from the pack) [10];
δ_2 = *cult* – *soc* (random coyote) [10].

Calculate,

$$new\_fit = f\left(new\_soc\right) \quad (5)$$

$$soc\left(t+1\right) = \begin{cases} new\_soc, \text{ if } new\_fit < fit \\ soc\left(t\right), \quad \text{otherwise} \end{cases} \quad (6)$$

- Birth and death within a pack.
  - ✓ When a new pup is born, its *soc* is calculated based on its parents.
  - ✓ If there is a single coyote in the pack with worse *adaptation*, it dies and the pup lives.
  - ✓ If there is more than one with a *worse adaptation*, the *oldest* dies and the pup lives.
  - ✓ Else the pup dies.
- Migration between packs.

  There is a probability that a coyote is evicted from a pack

$$P_e = 0.005\left(N_c^2\right) \quad (7)$$

  and if this happens, it exchanges places with a random coyote from another pack.
- The *age* of the coyotes is updated.

GN-model of the COA

The *Universal* GN-model proposed in [13] is used to describe the COA. The GN-model, shown in Fig. 1, could be applied for describing the above-presented metaheuristic algorithm only by altering the characteristic functions of the tokens.

Initially, the χ-token is in place l_1. The token's characteristic is

"*Metaheuristic algorithm parameters* (e.g., number of packs, number of coyotes, probability of leaving a pack, stopping criteria – a maximum number of objective function evaluations) *and problem parameters* (e.g., objective function, upper and lower bounds, problem dimension)".

Initially, the token ξ is in place l_{13}. The purpose of this token throughout the whole functioning of the GN-model is to collect information such as

"*Best individuals, corresponding objective function value, computational time and count of the objective function evaluations for each algorithm iteration*".

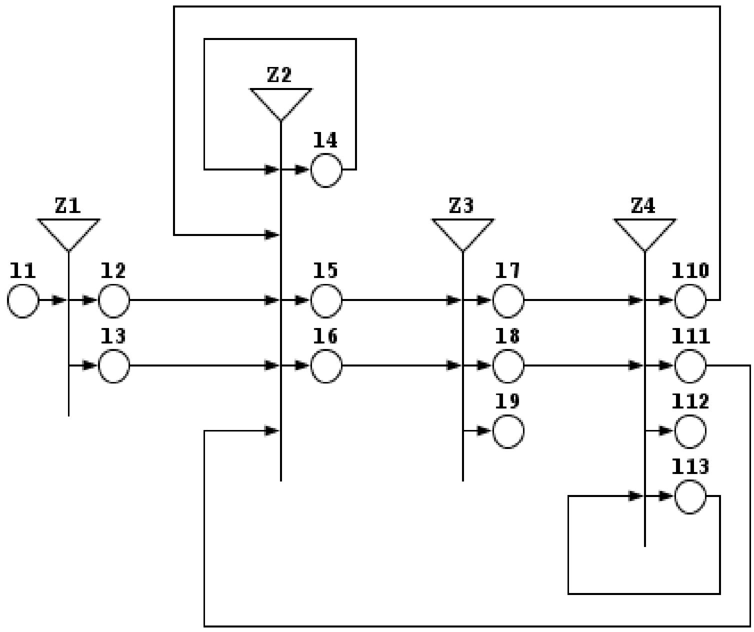

Fig. 1 Generalized net model of COA
(based on the *Universal GN-model*, proposed in [13])

The GN-transition Z_1, responsible for generating the initial population, has the form:

$$Z_1 = \langle \{l_1\}, \{l_2, l_3\}, r_1, l_1 \rangle,$$

$$r_1 = \begin{array}{c|cc} & l_2 & l_3 \\ \hline l_1 & True & True \end{array}.$$

The token χ splits into a δ- and a τ-token. Further, in place l_2, the δ-token obtains as a characteristic

"*Initial population*"

(using Eq. (1)).

In place l_3, the new token τ obtains the characteristic

"*Parameters of metaheuristic algorithm and parameters of the problem*".

The GN-transition Z_2, where the modification of the current generation takes place, has the form:

$$Z_2 = \langle \{l_2, l_3, l_4, l_{10}, l_{11}\}, \{l_4, l_5, l_6\}, r_2, \vee(\wedge(l_2, l_3), l_4, \wedge(l_{10}, l_{11})) \rangle,$$

$r_2 =$

	l_4	l_5	l_6
l_2	*True*	*False*	*False*
l_3	*True*	*False*	*True*
l_4	*False*	*True*	*False*
l_{10}	*True*	*False*	*False*
l_{11}	*True*	*False*	*True*

In place l_4, the tokens τ and δ unite into a new token, υ, with a characteristic

"*Generated modified population*".

During the functioning of this transition, based on COA metaheuristics, the initial population undergoes modifications aiming to produce better individuals. Here, the coyotes' adaptation is evaluated (Eq. (2)). Iteratively, the coyotes that belong to each pack are selected, alphas are detected according to the costs, the social tendency of the pack is computed, the social condition according to the alpha and the pack tendency is updated (Eq. (4)).

The new characteristic of the υ-token, in place l_5, is

"*New population*".

The τ-token, which comes from place l_3 or place l_{11}, obtains the characteristics

"*Metaheuristic algorithm parameters, problem parameters, current computational time, and the current count of the objective function evaluations*".

The third transition is responsible for ranking the solutions based on their performance and a predefined objective function, and consequently for determining the worst solutions. The GN-transition Z_3 has the form:

$$Z_3 = \langle\{l_5, l_6\}, \{l_7, l_8, l_9\}, r_3, \wedge(l_5, l_6)\rangle,$$

$r_3 =$

	l_7	l_8	l_9
l_5	*True*	*False*	*True*
l_6	*True*	*True*	*True*

The token υ splits into a ε- and a β-token. The characteristic of the ε-token (in place l_7) is

"*Rated individuals and their values of the objective function*".

The new social condition is evaluated (Eq. (5)), and based on Eq. (6) an adaptation is performed; a coyote can leave a pack and enter another pack (Eq. (7)).

The characteristic of the β-token (in place l_9) is

"*Worst individuals*",

the coyote with the worse adaptation in the pack dies.

When the token τ transfers from place l_6 to place l_8, only the count of the objective function evaluations and the computational time are updated in its characteristic.

Here, all individuals are ranked based on their performance and a predefined objective function.

The fourth transition of the GN-model should determine if the end of the algorithm has been reached. Z_4 has the form:

$$Z_4 = \langle\{l_7, l_8, l_{13}\}, \{l_{10}, l_{11}, l_{12}, l_{13}\}, r_4, \wedge(l_7, l_8, l_{13})\rangle,$$

$$r_4 = \begin{array}{c|cccc} & l_{10} & l_{11} & l_{12} & l_{13} \\ \hline l_7 & W_1 & F & W_2 & False \\ l_8 & W_1 & True & W_2 & False \\ l_{13} & False & False & False & True \end{array},$$

where W_1 = "The maximum number of the objective function evaluations has not been reached";
$W_2 = \neg\, W_1$.

In place l_{10}, the tokens ε and τ merge into a ς-token with a characteristic:

"*Rated individuals in the population and values of the objective function for the current iteration*"

if the algorithm has not finished yet. Finally, at the end of the algorithm, the token in place l_{12} obtains a new characteristic:

"*Best population, values of the objective function, total computational time and the count of the objective function evaluations*".

When the token τ transfers from place l_8 to place l_{11}, again only the sum of the objective function evaluations and the computational time in its characteristics are updated.

The best coyote (solution) with the corresponding value of the objective function, the computational time and the count of the objective function evaluations are added to the characteristic of the token ξ in place l_{13} at the end of each algorithm iteration. The information stored in the token ξ can be further considered a database.

Conclusion

In this paper, the generalized net theory is used to model the metaheuristics coyote optimization algorithm. The developed GN-model mimics the optimization processes based on the behaviour of *Canis latrans* species. The GN-model executes the COA conducting its basic steps and thus performing an optimal search.

The GN-model of COA allows the possibility to store information during its operation as a database. The collected data could be used in a case of comparable optimization problems as *a priori* information when choosing the COA parameters and/or for an approximate prediction of the outcomes. The use of the collected information in the database could serve to enhance the algorithm's performance.

The COA's GN-model considered here is another contribution to the open problem specified in [3], namely "to present each of the artificial intelligence areas by GNs". The presented work is a verification of the feasibility of converting the *Universal GN-model* proposed in [13] into the COA's GN model. The results serve to confirm the hypothesis that all concepts from artificial intelligence can be described by one uniform mathematical formalism – the generalized net theory.

Acknowledgements

The work presented here is partially supported by the National Scientific Fund of Bulgaria under Grant KP-06-N22/1 "Theoretical Research and Applications of InterCriteria Analysis".

References

1. Arfaoui J., H. Rezk, M. Al-Dhaifallah, M. N. F. Ibrahim, M. Abdelkader (2020). Simulation-based Coyote Optimization Algorithm to Determine Gains of PI Controller for Enhancing the Performance of Solar PV Water-pumping System, Energies, 13(17), https://doi.org/10.3390/en13174473.
2. Atanassov K. T. (1991). Generalized Nets, World Scientific, Singapore.
3. Atanassov K. T. (1998). Generalized Nets in Artificial Intelligence. Volume 1: Generalized Nets and Expert Systems, Prof. M. Drinov Publishing House, Sofia.
4. Bureva V., S. Popov, V. Traneva, S. Tranev (2019). Generalized Net Model of Cluster Analysis Using CLIQUE: Clustering in Quest, Int J Bioautomation, 23(2), 131-138.
5. Fidanova S., K. Atanassov, P. Marinov (2011). Generalized Nets in Artificial Intelligence: Generalized Nets and Ant Colony Optimization, Vol. 5, Prof. M. Drinov Publishing House, Sofia.
6. Fister I., I. Jr. Fister, X.-S. Yang, J. Brest (2013). A Comprehensive Review of Firefly Algorithms, Swarm and Evolutionary Computation, 13, 34-46.
7. Jamil M., X.-S. Yang (2013). A Literature Survey of Benchmark Functions for Global Optimization Problems, International Journal of Mathematical Modelling and Numerical Optimisation, 4(2), 150-194.
8. Lubich M., V. Andonov, A. Shannon, C. Slavov, T. Pencheva, K. Atanassov (2021). A Generalized Net Model of the Human Body Excretory System, Advances in Intelligent Systems and Computing, 1308, https://doi.org/10.1007/978-3-030-77716-6_17.
9. Nguyen T. T., T. D. Pham, L. C. Kien, L. Van Dai (2020). Improved Coyote Optimization Algorithm for Optimally Installing Solar Photovoltaic Distribution Generation Units in Radial Distribution Power Systems, Complexity, 2020, Article ID 1603802, https://doi.org/10.1155/2020/1603802.
10. Pierezan J., L. Dos Santos Coelho (2018). Coyote Optimization Algorithm: A New Metaheuristic for Global Optimization Problems, 2018 IEEE Congress on Evolutionary Computation, Rio de Janeiro, Brazil, 1-8, https://doi.org/10.1109/CEC.2018.8477769.
11. Qais M. H., H. M. Hasanien, S. Alghuwainem, A. S. Nouh (2019). Coyote Optimization Algorithm for Parameters Extraction of Three-diode Photovoltaic Models of Photovoltaic Modules, Energy, 187, 116001, https://doi.org/10.1016/j.energy.2019.116001.
12. Ribagin S., B. Zaharieva, I. Radeva, T. Pencheva (2018). Generalized Net Model of Proximal Humeral Fractures Diagnosing, Int J Bioautomation, 22(1), 11-20.
13. Roeva O., D. Zoteva, V. Atanassova, K. T. Atanassov, O. Castillo (2020). Cuckoo Search and Firefly Algorithms in Terms of Generalized Net Theory, Soft Computing, 24, 4877-4898.
14. Shehab M., A. T. Khader, M. A. Al-Betar (2017). A Survey on Applications and Variants of the Cuckoo Search Algorithm, Applied Soft Computing, 61, 1041-1059.
15. Sotirova E., H. Bozov, V. Stoyanov, I. Popov (2021). A Generalized Net Model of the Hybrid System for Diagnostics, Advances in Intelligent Systems and Computing, 1081, https://doi.org/10.1007/978-3-030-47024-1_15.

16. Yuan Z., W. Wanga, H. Wanga, A. Yildizbasi (2020). Developed Coyote Optimization Algorithm and Its Application to Optimal Parameters Estimation of PEMFC Model, Energy Reports, 6, 1106-1117.

Prof. Olympia Roeva, Ph.D.
E-mail: olympia@biomed.bas.bg

Olympia Roeva received M.Sc. Degree (1998) and Ph.D. Degree (2007) from the Technical University – Sofia. At present, she is a Professor at the Institute of Biophysics and Biomedical Engineering – Bulgarian Academy of Sciences. She has more than 100 publications, among those 10 books and book chapters, with more than 1000 citations. Her current scientific interests are in the fields of modelling, optimization and control of biotechnological processes, metaheuristic algorithms, intuitionistic fuzzy sets and generalized nets.

Assist. Prof. Dafina Zoteva, Ph.D.
E-mail: dafy.zoteva@gmail.com

Dafina Zoteva has a M.Sc. Degree in Bio- and Medical Informatics (2010) from Sofia University and Ph.D. Degree (2021) from the Institute of Biophysics and Biomedical Engineering at the Bulgarian Academy of Sciences. Currently, she is an Assistant Professor at the Faculty of Mathematics and Informatics at Sofia University. She has more than 30 publications. Her scientific interests are in the fields of metaheuristic algorithms, intuitionistic fuzzy sets and generalized nets.

Assoc. Prof. Peter Vassilev, Ph.D.
E-mail: peter.vassilev@gmail.com

Peter Vassilev has a M.Sc. Degree in Applied Mathematics (2009) from the Technical University – Sofia and Ph.D. Degree (2013) from the Institute of Biophysics and Biomedical Engineering, Bulgarian Academy of Sciences. At present, he is an Associate Professor at the Institute of Biophysics and Biomedical Engineering – Bulgarian Academy of Sciences. He has more than 40 publications. His current scientific interests are in the area of bioinformatics, artificial intelligence and algorithms.

© 2022 by the authors. Licensee Institute of Biophysics and Biomedical Engineering, Bulgarian Academy of Sciences. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).