



Relational Bayesian Optimization for Permutation

Bo-Wei Huang

Taiwan Evolutionary Intelligence Laboratory
National Taiwan University
r10921099@ntu.edu.tw

Hsu-Chen Liao

Taiwan Evolutionary Intelligence Laboratory
National Taiwan University
r10921118@ntu.edu.tw

Wen-Zhong Fang

Taiwan Evolutionary Intelligence Laboratory
National Taiwan University
r10921071@ntu.edu.tw

Tian-Li Yu

Taiwan Evolutionary Intelligence Laboratory
National Taiwan University
tianliyu@ntu.edu.tw

ABSTRACT

Relational Bayesian optimization for permutation (RBOP) is a new permutation estimation of distribution algorithm proposed in this paper. RBOP uses binary relations to represent the common property in permutations. Inspired by the Bayesian optimization algorithm, RBOP first builds a Bayesian network using binary relations. Then, RBOP samples genes using the most certain edge in the Bayesian network. In the scenario of black-box optimization, RBOP aims to solve various permutation problems with a limited number of function evaluations. Experiments show that in terms of average relative percentage deviation, RBOP outperforms edge histogram-based sampling algorithm on quadratic assignment problems, permutation flow shop problems and linear ordering problems. Additionally, RBOP also outperforms both node histogram-based sampling algorithm and kernels of Mallows model using Cayley distance on traveling salesman problems, permutation flow shop problems, linear ordering problems and 6 out of 10 instances of quadratic assignment problems.

CCS CONCEPTS

• Mathematics of computing → Evolutionary algorithms.

KEYWORDS

permutation, estimation of distribution algorithms, binary relation

ACM Reference Format:

Bo-Wei Huang, Wen-Zhong Fang, Hsu-Chen Liao, and Tian-Li Yu. 2023. Relational Bayesian Optimization for Permutation. In *Genetic and Evolutionary Computation Conference Companion (GECCO '23 Companion)*, July 15–19, 2023, Lisbon, Portugal. ACM, New York, NY, USA, 4 pages. <https://doi.org/10.1145/3583133.3590606>

1 INTRODUCTION

Permutation problems are commonly encountered in various fields, such as scheduling, resource allocation, DNA sequencing and more. In the past, some researchers attempt to use integer estimation of distribution algorithms (EDAs) to solve permutation problems [3].

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

GECCO '23 Companion, July 15–19, 2023, Lisbon, Portugal

© 2023 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-0120-7/23/07.

<https://doi.org/10.1145/3583133.3590606>

However, permutation has a unique property of mutual exclusivity constraints, which distinguishes it from the integer domain. Directly applying integer EDA to solve permutation problems is not the best solution.

In 2002, Tsutsui proposed a permutation EDA called the edge histogram-based sampling algorithm (EHBSA) [9] and in 2006 the same author proposed another permutation EDA called the node histogram-based sampling algorithm (NHBSA) [10]. These two algorithms perform well on the traveling salesman problem (TSP) and the quadratic assignment problem (QAP), respectively. Additionally, research [3] has shown that permutation EDAs like EHBSA and NHBSA which are specialized for permutations perform much better on other permutation problems than integer or real value EDAs because they do not encounter the same issues that integer EDAs face.

Afterward, the research of permutation EDAs have primarily deviated into two directions. In one of the directions, researchers firstly encode permutation chromosomes into random keys and then use the encoded chromosomes to solve permutation problems by EDAs. In the other direction, researchers incorporate probabilistic models commonly used in ranking fields into the EDA framework to create new permutation EDAs, which are then used to solve permutation problems.

In the research direction of using encoded chromosomes to solve permutation problems, Bosman *et al.* proposed the permutation gene-pool optimal mixing evolutionary algorithm (PGOMEA) [2] in 2016 and Wozniak *et al.* proposed P4 [11] in 2020. Both models achieve good results on the permutation flow shop problems (PFSP). The random key-based EDA (RKEDA) [1] proposed by M. Ayodele in 2018 achieved good results on various permutation problems.

In the research direction of incorporating ranking models into the EDA framework, Ceberio *et al.* incorporated the Plackett-Luce model into the EDA framework [6] and achieved good results on the linear ordering problems (LOP). Furthermore, Ceberio *et al.* also incorporated the Mallows model into the EDA framework [5]. In the Mallows model EDA (MMEDA) framework, different permutation distances, such as Cayley, Ulam, Kendall's- τ and Hamming, result in different performances on permutation problems with different semantics [4].

These previously proposed permutation EDAs show varying performance on problems with different semantics. Under the scenario of black-box optimization, this paper aims to break down the permutations into more fine-grained relations for adapting to the unique semantics of different problems. To achieve this goal, this

paper proposes a general EDA called relational Bayesian optimization for permutation (RBOP). Inspired by the Bayesian optimization algorithm (BOA) [8] in the binary domain, RBOP uses binary relations to represent the characteristic of the permutation problems and constructs a Bayesian network for information from promising chromosomes. The next generation of the population is sampled using the network.

The remainder of this paper is organized as follows. Section 2 details RBOP. Section 3 describes the experiments and demonstrates the results. Section 4 concludes this paper.

2 METHODOLOGY

In this section, we first introduce a common property in the permutation domain, which is represented using binary relation. Then, we present the framework of RBOP.

2.1 Binary relation

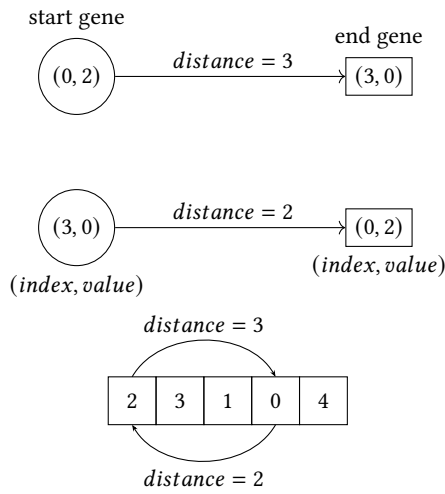


Figure 1: An example of using binary relation to represent the relation between genes.

In the permutation domain, there is a common property that can not only express the relative position between two elements and the distance between them, but also the absolute position of an element in the permutation. For instance, in a job scheduling problem, job A may need to be executed 3 jobs after job B. Furthermore, if we already know that job C is the 5_{th} job to be executed, and if job D needs to be executed 3 jobs after job C, then we can conclude that job D is the 8_{th} job to be executed.

RBOP decomposes permutations into the previously mentioned properties and express them using binary relations. If there is a start gene with value A and an end gene with value B , and the distance between A and B is d , then this relationship is expressed as (A, d, B) .

In addition, we use graphs to represent binary relations in the following sections, as shown in Figure 1. Each gene is represented as a node in the graph and is denoted by the tuple $(index, value)$ in Figure 1. The value of the distance between two genes is defined as

the number of positions the end gene is after the start gene. Additionally, we consider each chromosome to have circular properties for consistency in representation. The distance between two genes is formulated as follows:

$$distance = (end_{idx} - start_{idx}) \mod l,$$

where end_{idx} is the index of the end gene, $start_{idx}$ is the index of the start gene, l is the problem size.

2.2 The framework of RBOP

The framework of RBOP is similar to most EDA frameworks, consisting of the learning stage and the sampling stage. During the learning stage, RBOP learns the binary relations from the promising chromosomes and constructs a matrix to store the information.

In the sampling stage, RBOP starts by randomly selecting a parent chromosome from the population and deciding which genes to sample and which to inherit from the parent chromosome. The genes inherited from the parent chromosome serve as start genes to determine the values of genes that need to be sampled. After deciding which parent chromosome is used and which genes need to be sampled, RBOP builds a unique network using the parent chromosome and genes. Figure 3 is an example network constructed using the parent chromosome P_T and the binary relation matrix m in Figure 2.

After building the graph, RBOP uses the edge with the minimum entropy in the graph to sample the gene that needs to be sampled and determine the probability distribution of the possible gene values in a Bayesian manner. After sampling each gene, RBOP removes the edges connected to the previously sampled node in the graph and updates the entropy of the remaining edges.

3 TEST PROBLEMS AND EXPERIMENTS

In this section, the test problems used in this paper are introduced. Then, the experiment setup is detailed. Finally, the results between different models on four optimization problems and the discussion are presented.

3.1 Test problems

In this subsection, we introduce four widely used [1, 3, 10] permutation problems, namely TSP, QAP, PFSP and LOP, and their different problem characteristics.

TSP is a classic NP-Complete problem focusing on the adjacency relation of each element in the permutation. The objective of this problem is to find the shortest path that can visit every city exactly once.

QAP emphasizes the absolute position of each element in the permutation. The objective of QAP is to assign facilities to locations such that the sum of the products of the distances and the flow between each facility and location is minimized.

PFSP is a problem that aims to minimize the total time required to complete all jobs using a set of machines. The fitness value in PFSP depends on both the processing time of each job and the order of the previously scheduled jobs [4].

LOP is a maximization problem that the contribution of an index π_i to the fitness function depends on the previous and posterior indexes to it [3].

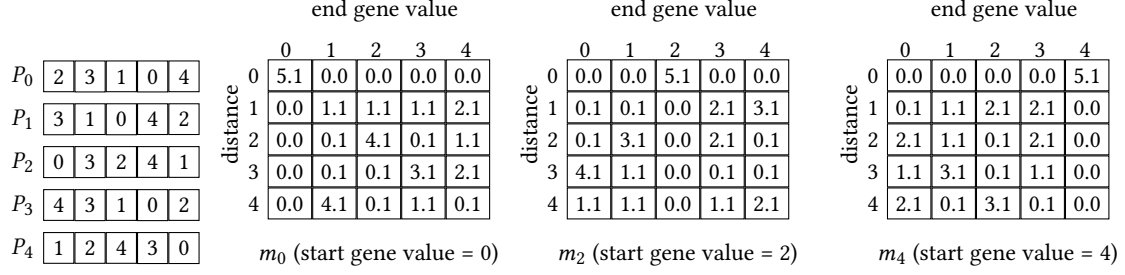


Figure 2: An example of a binary relation matrix, where P_0 to P_4 on the left represent the 5 chromosomes in the population, and the 3 matrices on the right are constructed from the population on the left. The bias constant 0.1 is added to the elements in the matrices.

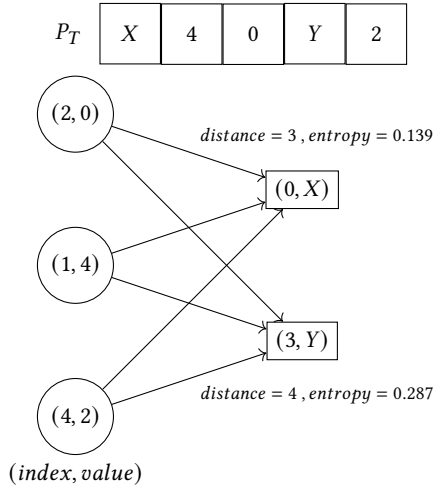


Figure 3: An example network constructed using the parent chromosome P_T and the binary relation matrix in Figure 2. The circular nodes represent the genes inherited from the parent chromosome, and the rectangle nodes represent the genes that need to be sampled. Gene X and gene Y in the parent chromosome are the genes that need to be sampled.

3.2 Experiment setup

The experiment aims to find the most general permutation EDA for solving the permutation problems. In the experiment, the performance of the algorithm is measured using the distance between the best solution found by the algorithm and the optimal solution of the problem instance, namely ARPD. The formula for ARPD is as follows:

$$ARPD = \left(\sum_{i=1}^n \frac{(Algorithm_i - Best) \times 100}{Best} \right) \times \frac{1}{n}, \quad (1)$$

where $Algorithm_i$ is the best solution found by the model each time, $Best$ is the optimal solution of the problem instance and n is the number of repetitions.

Three permutation EDAs are chosen to compare the performance with RBOP. Firstly, NHBSA and EHBSA are not only milestones

permutation EDAs but are also often used for comparison with other EDAs. Their performance is comparable to or even outperforms that of RKEDA [1], so NHBSA and EHBSA are compared with RBOP. KMMEDA is selected to compare the performance among all variants of MMEDA. In the black-box optimization scenarios, KMMEDA is incorporated with Cayley distance due to its relatively good performance compared to Kendall's- τ distance on both PFSP and QAP [7].

For fair comparison, we sweep the population size for each model to find the optimal population size for that model. For the problem instances, the experiment selects 10 instances from TSPLIB, QAPLIB, LOLIB, XLOLIB, and Taillard instances.

Table 3 shows the result of RBOP, EHBSA, NHBSA, and KMMEDA on TSP. In comparison to NHBSA and KMMEDA, RBOP produces solutions that are closer to the optimal solution in all TSP instances.

Table 1: The performance (ARPD) of RBOP, EHBSA, NHBSA, and KMMEDA on TSP.

TSP				
instance	RBOP	EHBSA	NHBSA	KMMEDA
gr24	1.447	0.000	3.121	13.365
gr48	6.916	0.000	20.914	33.845
eil51	6.244	0.000	21.479	27.606
berlin52	7.299	0.000	25.935	31.026
eil76	11.338	0.000	35.483	39.424
pr76	14.437	0.000	43.586	49.135
gr96	22.205	0.000	66.308	61.306
rat99	21.850	0.000	66.111	64.467
eil101	15.262	0.636	44.928	44.436
pr107	48.093	0.000	130.769	193.169

Table 4 shows the result of RBOP, EHBSA, NHBSA, and KMMEDA on QAP. For QAP, RBOP outperforms EHBSA in all instances, KMMEDA in 6 out of 10 instances, and even NHBSA in 6 out of 10 instances. This demonstrates that the absolute relation in the permutations can be effectively decomposed.

Tables 5 and 6 show the results of RBOP, EHBSA, NHBSA, and KMMEDA on PFSP and LOP. Although the problem characteristics of PFSP and LOP are not as clearly decomposable into smaller

Table 2: The performance (ARPD) of RBOP, EHBSA, NHBSA, and KMMEDA on QAP.

QAP				
instance	RBOP	EHBSA	NHBSA	KMMEDA
bur26a	0.001	0.267	0.010	0.221
bur26b	0.002	0.233	0.008	0.236
tai40a	3.169	6.701	3.216	2.904
tai40b	0.203	3.209	0.782	8.023
tai60a	4.073	7.165	3.781	3.060
tai60b	0.656	7.252	0.983	5.588
tai80a	4.059	6.449	2.799	2.447
tai80b	2.050	12.585	2.329	4.977
tai100a	4.343	6.282	2.812	2.307
tai100b	1.346	11.393	0.522	4.151

relations as TSP and QAP, RBOP finds the closest solutions in all instances of PFSP and finds the closest solutions in larger instances of LOP. This proves that even if the problem characteristics are not clear, RBOP can still solve permutation problems well by decomposing the problem into smaller relations.

Table 3: The performance (ARPD) of RBOP, EHBSA, NHBSA, and KMMEDA on PFSP.

PFSP				
instance	RBOP	EHBSA	NHBSA	KMMEDA
50_10_0	2.151	3.555	2.587	4.485
50_10_1	2.135	3.363	2.736	4.037
50_20_0	2.058	4.141	2.259	4.288
50_20_1	1.778	3.714	2.557	4.410
100_5_0	1.715	2.223	2.348	2.564
100_5_1	1.652	1.803	2.382	2.620
100_10_0	2.707	4.796	3.307	3.544
100_10_1	3.333	5.543	3.791	4.481
100_20_0	3.354	6.418	3.819	4.416
100_20_1	3.463	5.602	3.775	3.800

Table 4: The performance (ARPD) of RBOP, EHBSA, NHBSA, and KMMEDA on LOP.

LOP				
instance	RBOP	EHBSA	NHBSA	KMMEDA
t65b11xx	0.020	0.078	0.038	0.549
t65d11xx	0.006	0.061	0.004	0.255
t65f11xx	0.055	0.017	0.121	0.305
stabu1	0.037	0.010	0.020	0.531
stabu2	0.066	0.080	0.064	0.493
stabu3	0.149	0.169	0.149	0.403
N-t65b11xx_150	2.883	9.289	3.373	4.539
N-t65d11xx_150	4.302	10.748	4.716	5.595
N-t65f11xx_150	4.227	10.556	4.539	5.476
N-t65l11xx_150	1.637	6.247	2.010	2.756

4 CONCLUSION

RBOP proposed in this paper utilizes binary relations to adaptively represent the characteristic of the permutation problem and uses the relations as conditional probabilities to construct a Bayesian network from promising chromosomes. To distinguish the important relations from the others, RBOP uses Shannon's entropy to decide which relations are used for sampling new chromosomes. The experiment shows that RBOP obtains the lowest ARPD on average among TSP, QAP, PFSP, and LOP. Hence, when the problem is unknown, RBOP provides relatively closer solutions to the optimal compared to EHBSA, NHBSA, and KMMEDA for solving permutation problems. This paper demonstrates another effort to find a unified model for permutation problems, other than the Mallows distribution. As for future work, we would like to continue our investigation toward finding a more unified adaptive model for permutation problems with different semantics.

ACKNOWLEDGMENTS

The authors would like to thank the support by the National Science and Technology Council in Taiwan under grant No. NSTC 111-2221-E002-189 and National Taiwan University under grant No. 112L891103.

REFERENCES

- [1] M. Ayodele. 2018. *Effective and efficient estimation of distribution algorithms for permutation and scheduling problems*. Ph. D. Dissertation.
- [2] P. A. N. Bosman, N.H. Luong, and D. Thierens. 2016. Expanding from discrete cartesian to permutation gene-pool optimal mixing evolutionary algorithms. In *Proceedings of the Genetic and Evolutionary Computation Conference 2016*. 637–644.
- [3] J. Ceberio, E. Irurozki, A. Mendiburu, and J. A. Lozano. 2012. A review on estimation of distribution algorithms in permutation-based combinatorial optimization problems. *Progress in Artificial Intelligence* 1 (2012), 103–117.
- [4] J. Ceberio, E. Irurozki, A. Mendiburu, and J. A. Lozano. 2014. Extending distance-based ranking models in estimation of distribution algorithms. In *2014 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 2459–2466.
- [5] J. Ceberio, A. Mendiburu, and J. A. Lozano. 2011. Introducing the mallows model on estimation of distribution algorithms. In *Neural Information Processing: 18th International Conference, ICONIP 2011, Shanghai, China, November 13–17, 2011, Proceedings, Part II* 18. Springer, 461–470.
- [6] J. Ceberio, A. Mendiburu, and J. A. Lozano. 2013. The Plackett-Luce ranking model on permutation-based optimization problems. In *2013 IEEE congress on evolutionary computation*. IEEE, 494–501.
- [7] J. Ceberio, A. Mendiburu, and J. A. Lozano. 2015. Kernels of mallows models for solving permutation-based problems. In *Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation*. 505–512.
- [8] M. Pelikan, D. E. Goldberg, E. Cantú-Paz, et al. 1999. BOA: The Bayesian optimization algorithm. In *Proceedings of the genetic and evolutionary computation conference GECCO-99*, Vol. 1. Citeseer, 525–532.
- [9] S. Tsutsui. 2002. Probabilistic model-building genetic algorithms in permutation representation domain using edge histogram. In *Parallel Problem Solving from Nature—PPSN VII: 7th International Conference Granada, Spain, September 7–11, 2002 Proceedings*. Springer, 224–233.
- [10] S. Tsutsui. 2006. A comparative study of sampling methods in node histogram models with probabilistic model-building genetic algorithms. In *2006 IEEE International Conference on Systems, Man and Cybernetics*, Vol. 4. IEEE, 3132–3137.
- [11] S. Wozniak, M.W. Przewozniczek, and M.M. Komarnicki. 2020. Parameter-less population pyramid for permutation-based problems. In *Parallel Problem Solving from Nature—PPSN XVI: 16th International Conference, PPSN 2020, Leiden, The Netherlands, September 5–9, 2020, Proceedings, Part I* 16. Springer, 418–430.