Optimizing Permutation-Based Problems With a Discrete Vine-Copula as a Model for EDA

Abdelhakim Cheriet
Kasdi Merbah University
Ouargla, Algeria
abdelhakim.cheriet@univ-ouargla.dz

Roberto Santana
University of the Basque Country (UPV/EHU)
San Sebastian, Spain
roberto.santana@ehu.es

ABSTRACT

In this paper, we introduce a copula-based EDA that uses a Discrete Vine-Copula (DVC) model. This model is particularly suited to represent distributions in the permutation representation. To demonstrate the effectiveness of the proposed Discrete-Vine-Copula based EDAs (DVCEDA), we perform a set of experiments on instances of the known TSP problems. The results obtained are promising to extend the work on other class of problems.

CCS CONCEPTS

• **Mathematics of computing → Combinatorial optimization**; *Probabilistic algorithms*; *Discrete optimization*; • **Theory of computation → Discrete optimization**; **Evolutionary algorithms**.

KEYWORDS

Optimization, permutation-based problem, estimation of distribution algorithm, copula, vine-copula, evolutionary algorithm.

ACM Reference Format:

Abdelhakim Cheriet and Roberto Santana. 2019. Optimizing Permutation-Based Problems With a Discrete Vine-Copula as a Model for EDA. In *Genetic and Evolutionary Computation Conference Companion (GECCO '19 Companion), July 13–17, 2019, Prague, Czech Republic.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3319619.3321961

1 INTRODUCTION

Recently, estimation of distribution algorithms (EDAs), a class of EAs that learn a probabilistic model of the best solutions have been extended to deal with permutation-based problems [1]. In this paper, we investigate the used of copula-based models [5] for representing problems with a permutation representation. Copulas are functions that allow defining a joint probability distribution in terms of its univariate marginal distributions. This independence between the way marginal distributions are defined and the function (copula) used to specify the interaction between the marginal distributions, provides great flexibility for modeling.

Our EDA is based on a variant of vines that allows to model discrete variables [3]. Our goal is twofold, firstly, we want to determine whether copula-based models serve as an efficient approach to permutation-representation problems in comparison to other evolutionary algorithms. Secondly, we would like to know whether the promising results reported for copula-based EDAs in the continuous domain hold also in the permutation domain.

GECCO '19 Companion, July 13–17, 2019, Prague, Czech Republic

ACM ISBN 978-1-4503-6748-6/19/07...$15.00
https://doi.org/10.1145/3319619.3321961

2 DISCRETE VINE-COPULA (DVC)

Definition 2.1. A function $C(u, v) : [0, 1]^2 \to [0, 1]$ is a *copula* if and only if;

(1) For every $0 \leq u \leq 1$ $C(u, 0) = C(0, u) = 0$
(2) For every $0 \leq u \leq 1$ $C(u, 1) = u$ and $C(1, u) = u$
(3) For every $0 \leq u_1 \leq u_2 \leq 1$ and every $0 \leq v_1 \leq v_2 \leq 1$ $C(u_2, v_2) - C(u_2, v_1) - C(u_1, v_2) + C(u_1, v_1) \geq 0$

Copulas therefore satisfy the conditions of zero-grounded bivariate distribution functions of U and V with uniform margins. Hence a probabilistic interpretation may be given in the same way as any other joint cumulative distribution function (JCF) $C(u, v) = \Pr(U \leq u, V \leq v)$. Then the unique joint probability density function (JDF) $c(u, v)$ assocciated to C is such that $C(u, v) = \int_{-\infty}^{u} \int_{-\infty}^{v} c(v, v) dv dv$.

The relevance and utility of copulas are due to Sklar's theorem [5]. Thus, it is possible to separate the marginal behaviour due to the individual contributions of the random variables X, Y, described by its margins F_X and F_Y respectively, and the dependence structure, which is given by the copula C. Moreover, a key feature of copulas is that they are invariant under strictly monotone transformations of their random variables (U and V).

According the Sklar's theorem [5], every CDF F_X can be decomposed into margins $F_1, \cdots, F_d$ and a copula. Sklar's Theorem holds for mixed discrete and continuous distributions and thus provides a method to construct multivariate mixed distributions based on CDFs of copulas and margins[2].

A vine on n variables is a nested set of trees $T_1, \ldots, T_{n-1}$, where the edges of tree j are the nodes of tree $j + 1$ with $j = 1, \ldots, n - 2$. One of the special cases of vines is canonical vines (C-vines). In C-vines, in each tree T_j there is a unique node connected to $n - j$ edges. The C-vine density is given by

$$\prod_{k=1}^{n} f(x_k) \prod_{j=1}^{n-1} \prod_{i=1}^{n-j} c_{j,j+i|i,\ldots,j-1}, \tag{1}$$

For a vine with mixed margins, we sample from the corresponding continuous vine and apply the inversion method with the inverse of the margin cumulative distribution function to obtain mixed discrete and continuous samples. For mixed C-vine sampling, the authors of [2] use the algorithm for sampling from a continuous C-vine copula with uniform margins and then extend it by means of the inversion method to attach arbitrary continuous and discrete margins the detail of the algorithm can be found in [2]

3 DISCRETE VINE-COPULA BASED EDA

Our algorithm uses the mixed-vine copula model and incorporates a local optimization procedure for permutations, in particular it uses the 2-opt improvement method.The first step is the initialization of the initial population P_0 randomly. The next step is the selection step where we select the best individual.

The second main step is the modeling or the estimation of the distribution of the selected individuals. In this modeling step, we create a C-vine copula with mixed variables. The dimension of the vine equals the number of variables in the problem plus one, since we add one dimension for the value of fitness function. The root of the C-vine copula will be the value of the fitness function. The discrete part of the input variables is the elements of the permutation and the continuous part is the value of the fitness value. The vine will be truncated on the first level, this means that every position of an element in the permutation has a dependence with the global distance (fitness function value). For the generation step, the created C-vine will be used to generate the new solutions and we will drop the first value in every individual because it contains the generated sample of the function value. The result of the generation of permutations may contain repeated elements. Repeated elements are replaced by elements that are not present in the permutation in two ways: (1) Selecting one element not present at random.(2) Peeking the element that has the minimum distance with the element previous in the solution to the wrong element.

It is clear that the second way performs a local search. It takes an element from the wrong elements in the permutation then search for the best element in the adjacent of his predecessor (before it in the permutation). This way spends more time than the random one which we choose to use it. After this we perform a 2-opt local improvement of the generated solutions but not in every generation. The algorithm runs until the stop criterion is met. The last population found must contain the best solution found for problem.

4 EXPERIMENTS

We used the well-known benchmark suite TSPLIB95 [4] and we compared our DVC-EDA with a stat-of-art genetic algorithm and the Mallows Kernel EDA [1]. The parameters used by DVC-EDA (Population size,Sampled individuals,Max number of generations) are (100, 200,1000) while the The parameters for the GA (Population size, Crossover probability, Mutation probability, Selection method, Max number of generations) are (300, 0.7, 0.2, tournament, 1000), for the Mallows Kernel EDA, we've fixed the best parameters according to what is suggested in the original paper [1].

Table 1 shows the results of the algorithms. These results correspond to a single execution of each algorithm since we focus the comparison of the algorithms on the set of instances. It can be seen that our proposal outperforms the classical GA and MK-EDA in terms of fitness values. We also conducted a paired t-test for each pair of algorithms to evaluate for statistical differences. The results of the test showed that our algorithm outperforms the other two methods: for DCV-EDA vs MK-EDA, a significance level of $p = 0.0001$, and DCV-EDA vs GA ($p = 0.006$). There are also significant differences between GA and MK-EDA ($p = 0.006$).

Benchmarks	DCV-EDA	MK-EDA	GA	Best
kroA150	36390.0	219684	98299.0	26524
gr21	2707.0	5916	2998.0	2707
kroE100	27727.0	138997	55083.0	22068
att48	11363.0	124683	13574.0	10628
kroA200	42065.0	292663	154614.0	29368
gr17	2085.0	3489	2085.0	2085
berlin52	8229.0	24495	11055.0	7542
kroC100	26716.0	137783	55020.0	20749
gr229	191306.0	1227850	667642.0	134602
bier127	145403.0	569743	286247.0	118282
gr137	91551.0	549964	237314.0	69853
gr202	52321.0	245048	144091.0	40160
gr48	5397.0	16192	6318.0	5046
gil262	3522.0	23764	14116.0	2378
kroB150	36165.0	209851	111237.0	26130
gr24	1279.0	2771	1434.0	1272
burma14	3323.0	4350	3323.0	3323
a280	3958.0	31005	18062.0	2579
ch130	7787.0	39897	17467.0	6110
hk48	12345.0	40611	16361.0	11461
kroD100	26322.0	132911	50116.0	21294
kroA100	27800.0	136418	59190.0	21282
rat99	1517.0	7243	2364.0	1211
kroB200	43193.0	288230	145862.0	29437
gr96	64132.0	318695	125660.0	55209
ch150	8973.0	45913	25416.0	6528
kroB100	27432.0	132025	60696.0	22141

Table 1: Results of the algorithms on the TSP benchmarks

5 CONCLUSIONS

In this work, we have addressed the use of a C-vine copula as a model for EDAs. The C-vine copula used in this work is a mixed vine copula, that is able to represent interactions between continuous and discrete variables. Any candidate solution of the permutation-based problem is treated as the discrete variable, and the value of the fitness function is treated as the continuous variable. The experiments conducted have shown that the introduced algorithm produces good results compared to evolutionary algorithms, in particular the Mallows-model algorithm. One limitation of the C-vine modeling approach is that the learning methods it requires are hard to scale for large number of variables. This approach may not be suitable for problems with very large dimensions.

ACKNOWLEDGMENTS

R. Santana acknowledges support from the IT-609-13 (Basque Government) and TIN2016-78365-R (Spanish Ministry of Economy, Industry and Competitiveness) programs.

A. Cheriet acknowledges the computing resources provided on the HPC operated by the PTCI of Kasdi Merbah University Ouargla.

REFERENCES

[1] Josu Ceberio, Alexander Mendiburu, and Jose A. Lozano. 2015. Kernels of Mallows Models for Solving Permutation-based Problems. In *Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation (GECCO '15)*. ACM, New York, NY, USA, 505–512. https://doi.org/10.1145/2739480.2754741

[2] Arno Onken and Stefano Panzeri. 2016. Mixed vine copulas as joint models of spike counts and local field potentials. In *Advances in Neural Information Processing Systems*. 1325–1333.

[3] Anastasios Panagiotelis, Claudia Czado, and Harry Joe. 2012. Pair copula constructions for multivariate Discrete Data. *J. Amer. Statist. Assoc.* (2012).

[4] Gerhard Reinelt. 1995. TSPLIB95. *Interdisziplinäres Zentrum für Wissenschaftliches Rechnen (IWR), Heidelberg* 338 (1995).

[5] Abe Sklar. 1973. Random variables, distribution functions, and copulas. *Kybernetica* (1973), 449–460.