An Improved Co-evolutionary Decomposition-based Algorithm for Bi-level Combinatorial Optimization

Abir Chaabani
SOIE lab, University of Tunis, Tunisia
abir.chaabani@gmail.com

Slim Bechikh
SOIE lab, University of Tunis, Tunisia
slim.bechikh@gmail.com

Lamjed Ben Said
SOIE lab, University of Tunis, Tunisia
lamjed.ben.said@isg.rnu.tn

Radhia Azzouz
SOIE lab, University of Tunis, Tunisia
azzouzradhia@hotmail.com

ABSTRACT

Several real world problems have two levels of optimization instead of a single one. These problems are said to be bi-level and are so computationally expensive to solve since the evaluation of each upper level solution requires finding an optimal solution at the lower level. Most existing works in this direction have focused on continuous problems. Motivated by this observation, we propose in this paper an improved version of our recently proposed algorithm CODBA (CO-evolutionary Decomposition-Based Algorithm), called CODBA-II, to tackle bi-level combinatorial problems. Differently to CODBA, CODBA-II incorporates decomposition, parallelism, and co-evolution within both levels: (1) the upper level and (2) the lower one, with the aim to further cope with the high computational cost of the overall bi-level search process. The performance of CODBA-II is assessed on a set of instances of the MDVRP (Multi-Depot Vehicle Routing Problem) and is compared against three recently proposed bi-level algorithms. The statistical analysis of the obtained results shows the merits of CODBA-II from effectiveness viewpoint.

Categories and Subject Descriptors

I.2.8 [**Computing Methodologies**]: Articial Intelligence—*Problem Solving, Control Method, and Search.*

Keywords

Bi-level combinatorial optimization; co-evolution; decomposition; parallelism.

1. INTRODUCTION

Bi-level optimization is a branch of optimization where we find a nested optimization problem within the constraints of the outer one. In such kind of problems, we find a hierarchy between two optimization tasks. An interesting observation regarding the EBO (Evolutionary Bi-level Optimization) literature consists in that most works have tackled the continuous case. The number of EBO works for the discrete case is greatly reduced. Recently, to cope with the expensive cost of combinatorial BOPs, we have proposed CODBA [2] which is a search algorithm that decomposes the lower level population into a number of well-distributed populations that evolve in parallel while communicating between each others by means of co-evolution such that the information exchange is performed by means of recombination with best lower level solutions. CODBA has demonstrated a good performance on the bilevel MDVRP, which is a well-known combinatorial BOP. Motivated by this observation, we propose in this paper an improved version of CODBA, called CODBA-II which makes use of decomposition, parallelism, and co-evolution at both levels. The goal of CODBA-II is to further reduce the computational cost and to improve the quality of both lower level solutions and upper level ones.

2. AN IMPROVED CO-EVOLUTIONARY DECOMPOSITION BASED ALGORITHM

In this paper, we decompose the lower and upper level populations of CODBA-II into several well-distributed sub-populations over the whole level search space. Each sub-population could be seen as a cluster so that the clusters' centroids are well-distributed to cover as possible the whole search space. In this way, each sub-population is responsible for a specific region. All sub-populations co-evolve in parallel using several threads (one thread for each subpopulation). For brevity, we focus mainly on the algorithm description at the upper level. The algorithm details of the lower level is kept the same as the upper one. Other part of the algorithm like decomposition method is described in [2].

Step 1 (Initialization Scheme): We generate M_1 well distributed upper level sub-populations on the whole discrete decision space. To do this, we use a recently proposed DSDM method for discrete decision spaces that is described in [2]. Thereafter, the lower level optimization problem is executed to identify the optimal lower level solutions. In fact, the upper level fitness is assigned based on both the upper level function value and constraints.

Step 2 (Upper level parent selection): We choose $\frac{SPS_1}{2}$

GECCO '15 July 11-15, 2015, Madrid, Spain

ACM ISBN 978-1-4503-3488-4/15/07.
DOI: http://dx.doi.org/10.1145/2739482.2764701

Table 1: Average upper-level fitness value, Direct rationality value and Weighted rationality value obtained by CODBA-II, CODBA, COBRA and a reparing method for Bi-MDVRP intances. The symbol "+" means that $H_0$is rejected while the symbole "-" means the opposite. The best values are highlighted in bold.

	Upper level fitness				Lower level							
					Direct rationality				Weighted rationality			
instances	CODBA-II	CODBA	COBRA	Repair	CODBA-II	CODBA	COBRA	Repair	CODBA-II	CODBA	COBRA	Repair
birp01	**1794(--+)**	1821(++)	1890(-)	1869	0.8(+++)	1.7(++)	**0.7(+)**	5.2	11.9(+++)	18.6(++)	**11.7(+)**	3256.2
birp02	**3178(+++)**	3928(++)	4879(+)	3526	**0.9 (+++)**	2.1(++)	1.3(+)	4.6	**129.2(+++)**	139.2(++)	131.8(+)	4377.5
birp03	4245(+++)	4157(++)	8694(+)	**4083**	**2.7(+-+)**	3.3(-+)	2.8(+)	19.7	**280.3(+-+)**	310.3(-+)	284.6(+)	5183.1
birp04	**5838(+++)**	6820(++)	11234(+)	7355	**0.9(+-+)**	4.6(++)	1.2(+)	22.3	**190.6(+-+)**	265.7(++)	194.1(+)	5155.3
birp05	**6211(-++)**	6853(+-)	12142(+)	7991	0.9 (+++)	1.2(++)	**0.8(+)**	5.9	62.1(+++)	85.9(++)	**60.8(+)**	3291.8
birp06	**9582(-++)**	9612(+-)	16102(+)	10542	1.1(+++)	2.4(++)	**0.9(+)**	6.1	97.2(+++)	96.1(++)	**70.2(+)**	2413.7
birp07	**1494(+++)**	2171(++)	2988(+)	2651	2.9(+++)	**2.1 (++)**	3.3(+)	**1.9**	93.4(+++)	95.2(++)	98.8(+)	**85.1**
birp08	**3229(+++)**	4967(++)	8211(+)	6221	**3.4(+++)**	4.8(++)	3.7(+)	26.1	**208.3(+++)**	216.7(++)	209.9(+)	221.9
birp09	**4637(-++)**	5258(++)	13652(+)	7897	**3.8(+-+)**	9.3(++)	4.0(+)	28.3	**199.7(+-+)**	205.9(++)	195.2(+)	222.1
birp10	**9219(+++)**	11240(+-)	18657(+)	11502	0.95(+-+)	0.85(++)	**0.9(+)**	15.7	**75.5(+-+)**	78.4(++)	**71.3(+)**	118.2

population members from each upper level parent subpopulation using tournament selection where SPS_1 is the upper SubPopulation Size.

Step 3 (Variation at the upper level): Perform the crossover and mutation operations in order to create an offspring sub-population for each upper parent sub-population. We note that these operators are performed in parallel for the different upper sub-populations.

Step 4 (Lower level optimization): Solve the lower level optimization problem for each offspring.

Step 5 (Offspring evaluation): Combine each upper parent sub-population with its corresponding upper offspring population and evaluate them using the upper level objective function and constraints.

Step 6: (Environmental selection): Fill each new upper level sub-population using a replacement strategy. In fact, each new upper level sub-population is formed with the SPS_1 best solutions of the combined one. If the stopping criterion is met then store the best found upper level solution in the archive; otherwise, return to Step 2.

Step 7 (Co-evolution): Each sub-population member is crossed-over with one of the best archive members of the other sub-populations. In this way, we obtain an offspring population for each sub-population. Thereafter, we combine each sub-population with its corresponding offspring population and we update the sub-population by selecting the best SPS_1 ones. This process is repeated until the best lower level fitness function value is no more improved for a K generations or $MaxGenCoEvol$ is attained where $MaxGenCoEvol$ is the Maximum allowed number of Generations for Co-Evolution. Once the co-evolution is terminated, we return the global optimum to the upper level to evaluate upper level solutions.

3. RESULTS AND DISSCUSSION

Our experiments are divided into two parts. The first one is devoted to compare CODBA-II against three bi-level algorithms: (1) CODBA [2], (2) COBRA [3], and (3) Repairing method. The second one is dedicated to CPU time analysis to assess the efficiency of our CODBA-II from a computational time viewpoint. The different EAs were tested on the bi-MDVRP and the generated results are evaluated using three metrics: (1) Upper level fitness, (2) Direct rationality and (2) Weighted rationality [3]. As well, the performance comparison was carried out using the Wilcoxon statistical test. Therefore, thirty runs are perfomed for each couple (algorithm, problem). We observe from the Table 1 that CODBA-II seems competitive when compared to the contemporary approaches. In fact, it generates the better upper level fitness value regarding to the original CODBA, COBRA and the repairing approach. This result is explained by the fact that CODBA-II is based on the DSDM decomposition method [2] which can generate a well-diversified sub-populations that cover the whole decision space. Moreover, the co-evolution between those sub-populations can maintain a global view on the whole problem and can exploit the search capacity of each cluster. Regarding to the lower level solutions, we observe from Table 1 that CODBA-II improves the lower reactions regarding to CODBA in all instances. As well, the proposed scheme was able to outperform the repairing approach and COBRA algorithm in several instances. According to the CPU time comparision, we compute the consumed CPU Time in minutes for EAs on bi-pr01, bipr02 and bi-pr03 instances,we observe from the results that our CODBA-II consumes less CPU times than its competitors. In this regards, we can properly conclude that our proposed hierarchical CODBA-II is more adapted to the bi-level aspect of the problem.

4. CONCLUSION

In this paper, we have suggested an improved co-evolution ary algorithm for bilevel optimization to tackle bi-level combinatorial problems. The statistical analysis of the obtained results shows the merits of CODBA-II from effectiveness viewpoint.Therefore, it is interesting to design a multi-objective version of CODBA-II to solve the multi-objective bi-level optimization problem.

5. REFERENCES

[1] B. Colson, P. Marcotte, and G. Savard. Bilevel progra mming: A survey. 4OR, Springer, 3(2): 87–107, 2005.

[2] A. Chaabani, S. Bechikh, L. Ben Said. A Co-evolution ary Decomposition-based Algorithm for Bi-level Com binatorial Optimization. IEEE CEC, 2015 (accepted).

[3] F. Legillon, A. Liefooghe, and E-G. Talbi. CoBRA: A cooperative coevolutionary algorithm for bi-level optimization. IEEE CEC, 1–8, 2012.