An Approximate MIP-DoM Calculation for Multi-objective Optimization Using Affinity Propagation Clustering Algorithm

Claudio L. V. Lopes
Postgraduate Program in Mathematical and Computational Modeling, CEFET-MG
BH, MG, Brazil
claudiolucio@gmail.com

Flávio V. C. Martins
Computer Department at the CEFET-MG
BH, MG, Brazil 30510-000
flaviocruzeiro@cefetmg.br

Elizabeth F. Wanner
Computer Department at the CEFET-MG,
BH, MG, Brazil
efwanner@cefetmg.br

Kalyanmoy Deb
Department of Electrical and Computer Engineering
Michigan State University
Michigan, East Lansing, USA
kdeb@egr.msu.edu

ABSTRACT

Dominance move (DoM) is a quality indicator that compares two solution sets in a Pareto-optimal sense. The main issue related to DoM is its computational expense. A recent paper proposed a mixed-integer programming (MIP) approach for computing DoM that exhibited a computational complexity that is linear to the number of objectives and polynomial to the number of solutions. Even with this property, considering practical situations, the MIP-DoM calculation on some problems may take many hours. This paper presents an approximation method to deal with the problem using a cluster-based and divide-and-conquer strategy. Some experiments are tested, showing that the cluster based-algorithm is computationally much faster and makes a small percentage error from the original DoM value.

CCS CONCEPTS

• **Applied computing** → **Multi-criterion optimization and decision-making**;

KEYWORDS

Multi-objective optimization, Evolutionary multi-objective optimization, Computationally expensive optimization, Machine Learning, Cluster algorithms

ACM Reference Format:
Claudio L. V. Lopes, Flávio V. C. Martins, Elizabeth F. Wanner, and Kalyanmoy Deb. 2021. An Approximate MIP-DoM Calculation for Multi-objective Optimization Using Affinity Propagation Clustering Algorithm. In *2021 Genetic and Evolutionary Computation Conference Companion (GECCO '21 Companion), July 10–14, 2021, Lille, France.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3449726.3459445

GECCO '21 Companion, July 10–14, 2021, Lille, France

ACM ISBN 978-1-4503-8351-6/21/07.
https://doi.org/10.1145/3449726.3459445

1 INTRODUCTION

Dominance move (DoM) is a binary quality indicator used in multi-objective and many-objective optimization to compare two solution sets obtained from different algorithms [2]. The DoM measures the minimum 'effort' that one solution set has to make in trying to dominate the second set, precisely the minimum absolute sum of the movements in objective directions needed to make the first set dominant. Compared to ϵ-indicator, it has the advantage of considering all solutions of both sets. It also does not require any reference, such as HV and IGD+.

In a recent paper, a mixed-integer programming (MIP) approach is proposed to calculate DoM [1]. The calculation time is polynomial to the cardinality of sets and linear to the number of objectives.

There is still space for some improvements. The first work's contribution is related to clustering the solution sets. We use the affinity propagation [3] algorithm. The second contribution is assigning the clusters from sets, considering that MIP-DoM is a binary indicator. The final contribution shows a two-phase approach in which MIP-DoM approximation maintains the original MIP-DoM values and properties. It can generate values with an estimated error of less than 0.40% on average and a reduction in time spent by up to 5400 times, considering the experiments performed.

2 MIP-DOM QUALITY INDICATOR

Dominance move is a measure for comparing two solution sets, being classified as a binary indicator. Consider that $\boldsymbol{P}$ and $\boldsymbol{Q}$ are sets of points, with points $\boldsymbol{p}_i$, $i \in \{1, \ldots, |\boldsymbol{P}|\}$ and points $\boldsymbol{q}_j$, $j \in \{1, \ldots, |\boldsymbol{Q}|\}$. The dominance move of $\boldsymbol{P}$ to $\boldsymbol{Q}$, $DoM(\boldsymbol{P}, \boldsymbol{Q})$, is the minimum total distance of moving points of $\boldsymbol{P}$ to $\boldsymbol{P'}$, such that each and every point in $\boldsymbol{Q}$ is dominated by at least one point in $\boldsymbol{P'}$. The problem aims to obtain $\boldsymbol{P'} = \{\boldsymbol{p}'_1, \boldsymbol{p}'_2, \ldots, \boldsymbol{p}'_{|\boldsymbol{P}|}\}$ from $\{\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_{|\boldsymbol{P}|}\}$ such that $\boldsymbol{P'}$ dominates $\boldsymbol{Q}$ and the total move from $\{\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_{|\boldsymbol{P}|}\}$ to $\{\boldsymbol{p}'_1, \boldsymbol{p}'_2, \ldots, \boldsymbol{p}'_{|\boldsymbol{P}|}\}$ must be minimum.

The formal expression of DoM can be stated as [1]:

$$DoM(\boldsymbol{P}, \boldsymbol{Q}) = \min_{\boldsymbol{P'} \prec \boldsymbol{Q}} \sum_{i=1}^{|\boldsymbol{P}|} d(\boldsymbol{p}_i, \boldsymbol{p}'_i), \tag{1}$$

in which $d(\boldsymbol{p}_i, \boldsymbol{p}'_i)$ can be the Euclidean or the Manhattan distance.

A mixed-integer programming (MIP) approach is applied to calculate DoM [1]. However, the approach took a few minutes to several hours in different pairs of sets.

3 THE APPROXIMATION ALGORITHM

Our algorithm proposes to calculate MIP-DoM($\boldsymbol{P}$, $\boldsymbol{Q}$) in a two-phase manner. In Phase 1, the first step is to apply the clustering algorithm in both sets. The next step is to assign the cluster elements, creating pairs of clusters from $\boldsymbol{P}$ and $\boldsymbol{Q}$. The original MIP-DoM [1] is calculated for each assigned pair of clusters. In phase 2, the $\boldsymbol{P}$ solution set is substituted by the $\boldsymbol{P'}$ coming from all MIP-DoM cluster executions, and an approximate MIP-DoM is used with an additional parameter, the intermediary value, indicated by $\boldsymbol{D}$. Considering the MIP-DoM model presented in [1], the difference to the approximate MIP-DoM is the distance $\boldsymbol{D}$ (penalization strategy) added to the objective function. Algorithm 1 outlines the idea.

Algorithm 1 Approximate MIP-DoM with affinity propagation

```
1:  function MIP-DoM Affinity propagation(P,Q,preference,λ,T)
2:      Input: P,Q,preference,λ,T
3:      Output: ApproximateValue
4:                                                          ▷ PHASE 1
5:                                          ▷ calculating the similarity matrix
6:      S_P ← SIMILARITYMATRIX(P)
7:      S_Q ← SIMILARITYMATRIX(Q)
8:                                  ▷ generating the clusters from P and Q
9:      P clusters ← AFFINITY PROPAGATION(S_P, λ, preference, T)
10:     Q clusters ← AFFINITY PROPAGATION(S_Q, λ, preference, T)
11:                                       ▷ Assigning clusters using the ideal points
12:     Assignment ← ASSIGNMENT(P clusters, Q clusters)
13:                                  ▷ calculating the MIP-DoM for each assigned cluster
14:     P' ← ∅
15:     D ← ∅
16:     for each {(p_c, q_c)} ∈ Assignment do
17:         D_c, P'_c ← MIPDoM(p_c, q_c)
18:         P' ← P' ⋃ P'_c
19:         D ← D ⋃ D_c
20:                                                         ▷ PHASE 2
21:                          ▷ final calculation using the approximate MIP-DoM
22:     ApproximateValue ← APPROXIMATEMIPDoM(P', Q, D)
23:     return ApproximateValue
```

The $\boldsymbol{P}$ and $\boldsymbol{Q}$ solution sets are the first two input parameters, and the *preference* (used to control the number of clusters), λ (damping factor used to avoid numerical oscillations), and T (number of iterations) are the last ones, and related to affinity propagation clustering.

The first step is related to the similarity matrix calculation: *SIMILARITYMATRIX*. Then, we create the clusters using the affinity propagation *AFFINITYPROPAGATION*. The damping factor λ is 0.9, the *preference* is the median similarity value, and T is 200.

The next step is how to assign the clusters from $\boldsymbol{P}$ to every cluster from $\boldsymbol{Q}$. The *ASSIGNMENT* method has two parameters: clusters from $\boldsymbol{P}$ and $\boldsymbol{Q}$ which are formed by the elements in each cluster.

For each assignment, it is possible to execute the model to calculate the MIP-DoM value: *MIPDoM* function in line 17 Algorithm 1, as in [1]. D_c is the intermediary MIP-DoM value, and $\boldsymbol{P'_c}$ are the solution sets generated by MIP-DoM.

The approximate MIP-DoM step involves the use of $\boldsymbol{D}$ values, and $\boldsymbol{P'}$ solution set. The cardinality of $\boldsymbol{P'}$ is much smaller than $\boldsymbol{P}$ [1]. The *APPROXIMATEMIPDoM* differ from *MIPDoM* function in line 17, just by the $\boldsymbol{D}$ vector added in the objective function.

4 EXPERIMENTS

The algorithm validation method involves some multi-objective benchmark problems. These problem test sets have three objectives and come from *DTLZ* and *WFG* families. Some algorithms are used to generate the approximated Pareto front and using the outcomes in the MIP-DoM indicator.

In Table 1, the experimental results are presented. The columns labeled as 'Original' means the value obtained from the MIP-DoM, and the 'Approx.' represents the result obtained from our MIP-DoM approximation using affinity propagation. In the same manner, the time spent by each execution is also presented.

Table 1: Original MIP-DoM and Approximate MIP-DoM values and time spent in seconds for *DTLZ* and *WFG* families.

Problem	Algorithms	MIP-DoM		Time spent (s)	
		Original	*Approx.*	*Original*	*Approx.*
DTLZ2	*MOEA/D*	0.970	0.970	101138.66	**65.77**
	IBEA	1.000	1.000	976980.85	**286.50**
	NSGA-III	1.004	1.004	190875.98	**288.71**
	NSGA-II	1.013	1.013	39250.80	**253.25**
	SPEA2	1.022	1.022	107129.99	**72.73**
DTLZ7	*NSGA-III*	1.402	1.402	3345.07	**364.33**
	IBEA	1.468	1.508	65875.00	**135.76**
	MOEA/D	1.695	1.695	2520.99	**23.90**
	NSGA-II	1.782	1.791	8260.20	**363.03**
	SPEA2	2.184	2.184	2040.71	**21.74**
WFG1	*IBEA*	1.230	1.230	60.27	**22.33**
	MOEA/D	1.557	1.557	158.81	**36.62**
	SPEA2	1.659	1.659	341.12	**78.79**
	NSGA-III	1.822	1.822	950.14	**126.53**
	NSGA-II	1.944	1.944	184.51	**60.14**
WFG2	*IBEA*	1.503	1.503	283.05	**39.53**
	NSGA-III	1.571	1.571	181.02	**50.57**
	MOEA/D	1.683	1.683	328.40	**17.25**
	NSGA-II	1.687	1.687	140.20	**50.57**
	SPEA2	1.859	1.859	879.41	**24.47**

5 CONCLUDING REMARKS

The empirical results have shown that the proposed approximation method is faster than the original MIP-DoM calculation procedure, from $\sim$ 9 to $\sim$ 5, 400 times better, and the average approximation error is only 0.40%. This MIP-DoM approximation drastically reduces its computational cost, thereby making it applicable and useful to the EMO community.

REFERENCES

[1] Claudio Lucio do Val Lopes, Flávio Vinícius Cruzeiro Martins, Elizabeth Fialho Wanner, and Kalyanmoy Deb. 2020. Analyzing Dominance Move (MIP-DoM) Indicator for Multi- and Many-objective Optimization. *Submitted for consideration for publication in the IEEE Transactions on Evolutionary Computation* (2020). arXiv:cs.NE/2012.11557

[2] Miqing Li and Xin Yao. 2017. Dominance Move: A Measure of Comparing Solution Sets in Multiobjective Optimization. *CoRR* abs/1702.00477 (2017). arXiv:1702.00477 http://arxiv.org/abs/1702.00477

[3] L. Wang, Z. Hao, and W. Sun. 2019. A Novel Self-Adaptive Affinity Propagation Clustering Algorithm Based on Density Peak Theory and Weighted Similarity. *IEEE Access* 7 (2019), 175106–175115. https://doi.org/10.1109/ACCESS.2019.2956963