Co-evolution of Strategies for Multi-objective Games under Postponed Objective Preferences

Erella Eisenstadt
School of Mechanical Engineering
Tel Aviv University
Tel Aviv, Israel
and the Department of Mechanical Engineering
ORT Braude College of Engineering
Karmiel, Israel
Email: erella@braude.ac.il

Amiram Moshaiov
School of Mechanical Engineering
Tel Aviv University
Tel Aviv, Israel
Email: moshaiov@eng.tau.ac.il

Gideon Avigad
Department of Mechanical Engineering
ORT Braude College of Engineering
Karmiel, Israel
Email: gideona@braude.ac.il

***Abstract*—The vast majority of studies that are related to game theory are on Single Objective Games (SOG), also known as single payoff games. Multi-Objective Games (MOGs), which are also termed as multi payoff, multi criteria or vector payoff games, have received lesser attention. Yet, in many practical problems, generally each player cope with multiple objectives that might be contradicting. In such problems, a vector of objective functions must be considered. The common approach to deal with MOGs is to assume that the preferences of the players are known. In such a case a utility function is used, which transforms the MOG into a surrogate SOG.**

This paper deals with non-cooperative MOGs in a non-traditional way. The zero-sum MOG, which is considered here, involves two players that postponed their objective preferences, allowing them to decide on their preferences after tradeoffs are revealed. To solve such problems we propose a co-evolutionary algorithm based on a worst-case domination relation among sets. The suggested algorithm is tested on a simple differential game (tug-of-war). The obtained results serve to illustrate the approach and demonstrate the applicability of the proposed co-evolutionary algorithm.

I. Introduction

Many practical problems, in economics and engineering, can be modelled and solved using game-theoretic methods. Generally in such problems, each player may consider multiple objectives, which could be contradicting. In such problems, e.g., [1], a vector of objective functions must be considered. If players in a Multi-Objective Games (MOGs) have preferences toward the objectives then a utility function can be used to transform the MOG into a game with a single objective. Often, however, DMs would like to explore different alternatives before deciding on their preference of objectives. According to [2], in such a case, one needs to carry out *"the impossible task of solving all possible scalarizations."* From a decision support viewpoint, the current utility-based approaches to MOGs lack the insight that may be gained by revealing the objective tradeoffs.

The motivation for this work, which follows that of [3], is the desire to find all rational strategies, in order to take into considerations the trade-offs of performances, before making a decision on the preferences towards the objectives.

In [3], a novel multi-payoff game approach has been proposed, which allows such tradeoff-analysis before a decision is made on the selected strategy. The suggestion in [3] has recently been formalized in [4]. In [3] the problem was solved using a one-sided perspective method. In contrast, the current study suggests a co-evolutionary algorithm to solve such games from a two-sided perspective approach.

The rest of this paper is organized as follows. Section II provides the relevant background, and section III includes problem definition and solution approach. The co-evolutionary algorithm is described in section IV. The example and results are given in section V. Finally the conclusions are provided in section VI.

II. Background

This section describes some basic ideas which are relevant to the understanding of this paper. First, in section II-A definitions and notations are provided for several types of domination relations and in particular for worst-case domination. This is followed by a brief overview on MOGs in section II-B.

A. Domination Relations

Pareto optimality is the common approach to deal with multi-objective optimization when tradeoffs are sought before objective preferences are made (e.g., [5]). Pareto optimality is based on a domination relation among vectors.

1) Domination Relations among vectors:

Without loss of generality, the concept of domination among vectors for a maximization problem is defined as follows:
A vector $x \in \Upsilon$ dominates a vector $y \in \Upsilon$ ($x \succ y$), if $f^{(k)}(x) \geq f^{(k)}(y)$ for all $k \in [1, K]$ and there exists $k \in [1, K]$ for which $f^{(k)}(x) > f^{(k)}(y)$. A vector $x^* \in \Upsilon$ is called Pareto-optimal if no solution exists in Υ that dominates x^*. The set of all non-dominated solutions is the Pareto set $X^* := \{x \in \Upsilon : \neg\exists y \in \Upsilon : y \succ x\}$. Its image in the objective space, is called the Pareto front. To distinguish between minimization and maximization, the following notations are made. When vector $x \in \Upsilon$ dominates a vector $y \in \Upsilon$ it is denoted as $x \overset{max}{\succ} y$ for maximization and

978-1-4799-8622-4/15/$31.00 ©2015 IEEE

as $x \overset{min}{\succ} y$ for minimization. The meaning of $x \overset{min}{\succ} y$ is that $f^{(k)}(x) \leq f^{(k)}(y)$ for all $k \in [1, K]$ and there exists $k \in [1, K]$ for which $f^{(k)}(x) < f^{(k)}(y)$.

2) Domination Relations among sets:

Let $X = [x_1, x_2 \dots x_n]^T \in \Upsilon$ and $Y = [y_1, y_2 \dots y_n]^T \in \Upsilon$ be two sets. Following [6], set X dominates set Y in a maximization problem, is defined as follows:

$$X \overset{max}{\succ} Y \quad := \{\forall y \in Y \;\; \exists\, x \in X \mid x \overset{max}{\succ} y\} \tag{1}$$

In a minimization problem, the $\overset{max}{\succ}$ notation should be replaced by $\overset{min}{\succ}$.

According to [7] a set $Z^* \in \Upsilon$ is labeled Pareto-optimal if no other set exists in Υ that dominates set Z^*, and the set of all non-dominated sets is the Pareto set of sets. In the case of maximization the Pareto set of sets is defined as:

$$P^* := \{X \in \Upsilon : \neg \exists Y \in \Upsilon : Y \overset{max}{\succ} X\} \tag{2}$$

In the case of minimization, the same definition will be used but $\overset{max}{\succ}$ will be replaced by $\overset{min}{\succ}$. The image of the set P^* in the objective space is called the Pareto-layer [7].

3) Worst-case domination among sets:

The relation of worst-case domination is a particular relation type among sets, which has been used to find robust solutions to MOPs involving uncertainties (e.g., [8]). In the current paper, worst-case optimization is used for the solution of MOGs under postponed objective preferences. Let $X = [x_1, x_2 \dots x_n]^T \in \Upsilon$ and $Y = [y_1, y_2 \dots y_n]^T \in \Upsilon$ be two sets. Without loss of generality, for a maximization problem, the anti-optimal subset of the set X is the Pareto set of a *minimization* problem:

$$X^{-*} := \{x \in \Upsilon : \neg \exists x' \in \Upsilon : x' \overset{min}{\succ} x\} \tag{3}$$

This subset contains all the worst-case solutions of set X. A set $X \in \Upsilon$ worst-case dominates a set $Y \in \Upsilon$ in a maximization problem iff:

$$X \overset{wc}{\succ} Y \Longleftrightarrow Y^{-*} \overset{min}{\succ} X^{-*} \Longleftrightarrow Y \overset{min}{\succ} X \tag{4}$$

To distinguish between maximization and minimization, the following notations are made. When set X worst-case dominates set Y, ($X \overset{w.c}{\succ} Y$) it is denoted as $Y \overset{min}{\succ} Z$, for maximization , and as $Y \overset{max}{\succ} X$, for minimization.

B. Multi-Objective Games

The results of a MOG may be represented as a payoff vector for each player in a K dimensional space, whose K coordinates are the game's objectives. If the game is zero-sum, for each objective, the payoff of one player will have the same value as that of the other player, but with the opposite sign. Most studies that deal with MOGs involve the aggregation of payoffs by a weighted sum approach (e.g., [1], [2], [9], [10]). This is in contrast to the approach taken here, which follows the work in [3], [4], [11]–[13]. The papers [11]–[13] deal with MOGs using domination relations. Yet, in these studies the set of performance vectors, which is associated with a strategy is transformed into a unique vector, namely the nadir-point of the set. This is in contrast to [4], [3] and the current work, which employ a worst-case optimization approach. In [3] the game is solved by finding the rational strategies from the perspective of one player only. In contrast, here following [4], a MOG with postponed objective preference is considered from both players' perspectives.

III. Fundamentals

This section is based on previous studies including [3], [11], [12] and in particular [4].

A. Problem definition

In this work a game between two players, P_1 and P_2, is considered. Each player may choose a strategy out of a set of strategies. Each strategy is a decision vector such that the i^{th} strategy of P_1 is: $s_1^i = [s_1^{i(1)}, \dots, s_1^{i(n_1)} \dots, s_1^{i(N_1)}]^T \in \mathbb{R}^{N_1}$ the j^{th} strategy of P_2 is: $s_2^j = [s_2^{j(1)}, \dots, s_2^{j(n_2)} \dots, s_2^{j(N_2)}]^T \in \mathbb{R}^{N_2}$, where N_1, and N_2 are the corresponding number of strategy's decision parameters (see example in [3]).

Let S_1 and S_2 be the sets of all possible strategies of the two players respectively, such that: $S_1 = \{s_1^1, s_1^2 \dots, s_1^i, \dots, s_1^I\}$ and $S_2 = \{s_2^1, s_2^2 \dots, s_2^j, \dots, s_2^J\}$ where $S_1 \in \Omega \subseteq \mathbb{R}^{I \times N_1}$ and $S_2 \in \Gamma \subseteq \mathbb{R}^{J \times N_2}$, where I and J are the total number of strategies of P_1 and P_2, respectively. The interaction between the i^{th} strategy and the j^{th} strategy played by P_1 and P_2, respectively, results in a game $g_{i,j}$:

$$g_{i,j} := \{s_1^i, s_2^j\} \in \Phi \subseteq \mathbb{R}^{N_1} \times \mathbb{R}^{N_2} \tag{5}$$

All of the alternative interactions, among all strategies of both players, form the set G of all possible games:

$$G := \{g_{i,j} \in \Phi \mid s_1^i \in S_1 \;\; \wedge \;\; s_2^j \in S_2\} \tag{6}$$

Each game is evaluated using an objective vector of performances (payoff vector), $\bar{f} = [f^{(1)}, f^{(2)}, \dots, f^{(K)}]^T \in \Psi \subseteq \mathbb{R}^K$. More specifically, the result of the $g_{i,j}$ game is assessed by:

$$F(g_{i,j}) = \bar{f}_{i,j} = [f_{i,j}^{(1)}, f_{i,j}^{(2)}, \dots, f_{i,j}^{(K)}]^T \in \Psi \tag{7}$$

In general, for zero-sum games one player aims at minimizing the payoff (minimizer) and the other at maximizing the same payoff (maximizer). This can be stated as:

$$min/max \;\; F(g_{i,j}) \qquad \text{s.t} \quad g_{i,j} \in G \tag{8}$$

The optimization problem in equation 8 is ill-defined since that min or max operation, on vector F, has to be defined. First, each player must take into account the alternatives of the opponent. This leads to the need to consider two different optimization problems. The first problem is based on the minimizer (P_1) perspective, which aims at minimizing the objectives while considering the best opponent (the maximizer) strategies. This problem can be defined as a MinMax problem. The second problem is based on the maximizer (P_2) perspective, which aims at maximizing the objectives while considering the best opponent (the minimizer) strategies. This leads to a MaxMin problem. These are formulated as:

$$\text{For the minimizer } P_1 : \quad \min_{s_1^i \in S_1} \max_{s_2^j \in S_2} \bar{f}_{i,j} \tag{9}$$

$$\text{For the maximizer } P_2 : \quad \max_{s_2^j \in S_2} \min_{s_1^i \in S_1} \bar{f}_{i,j} \tag{10}$$

A game against an opponent with postponed objective preference is a game of incomplete information, as both players do not know how the opponent evaluates each objective. In such a MOG, if the players are "conservative" and do not wish to take any risks, they must assume that the worst-case will occur, hence, they will try to bound their payoff losses. For each player, in the zero-sum game, the problem can be defined as a worst-case optimization problem (as detailed in [4]). Each player considers the best performances of the opponent, which are its' own worst-cases. This is a conservative approach to deal with a game against an unpredictable opponent. When players solve either equation 9 or equation 10 their rational strategies, under the incomplete information, are the worst-case strategies, which are hereby termed rational strategies. The rational strategies of the minimizer (equation 9), include all his alternative strategies, which guarantee that whichever strategy the maximizer chooses, there will never be another minimizer's strategy that offers better performances in both objectives with respect to minimization. And vice versa for the maximizer's problem (equation 10).

B. Solution approach

As discussed in the above section, the set of performances associated with each strategy includes all the results of the interactions between the strategy and all the opponent's possible strategies. Following the suggested procedure in [4], this section provides some insight to the proposed algorithm (see section IV) using an introductory example. It concerns a MOG between two players, P_1 (minimizer) and P_2 (maximizer) where each player has four strategies ($I = J = 4$), and therefore there are sixteen possible games ($I \times J = 16$). The payoff vectors, from all interactions are illustrated on the left side of Figure 1, in which the strategies of P_1 are denoted by four different shapes, s_1^1-triangle, s_1^2-square, s_1^3-star and s_1^4-diamond. The strategies of P_2 are denoted by color-fillings, s_2^1-white, s_2^2-gray, s_2^3-black and s_2^4-dots. Without loss of generality, the steps of the search are to be demonstrated for the maximizer only.

The sorting of the strategies is based on the evaluation of the set of its payoff-vectors. The evaluation process involves three steps. The first step, which is described in subsection III-B1, is to find, for each strategy, the sub-set that represents the strategy performances. The second and third steps are to assign the representative sub-set with rank and in-rank grades, as described in subsections III-B2 and III-B3 respectively.

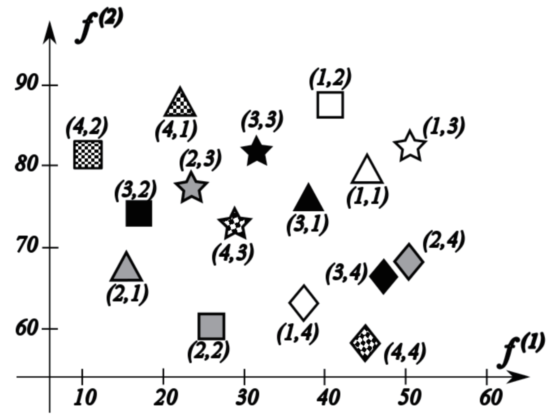

Fig. 1: Results of 16 games between P_1 and P_2

1) Finding the representative sub-set of a strategy: In considering the j-th strategy of P_2 (maximizer), I finite possible strategies of P_1 (minimizer) must be considered. All possible games, which involve the j-th P_2 strategy, form a set of I games that are associated with the P_2 j-th strategy: $G_{s_2^j} = \{g_{1,j}, g_{2,j}, \ldots, g_{i,j}, \ldots, g_{I,j}\} \subset G$. For example, in Figure 2, $G_{s_2^1}$ is the set of the four white shapes (labelled by (1,1), (2,1), (3,1) and (4,1)). The minimizer (P_1) i-th strategy aims at minimizing the objectives ($f^{(1)}$ and $f^{(2)}$) while playing with the maximizer (P_2) j-th strategy. This means that P_1 aims at:

$$\min_{s_1^i \in S_1} \bar{f}_{i,j} \ \forall i = 1, ..., I \tag{11}$$

For the maximizer j-th strategy, a set exists of minimizer strategies that may serve as solutions to equation 11. As the minimizer (P_1) aims at worst-case for the maximizer (P_2), these strategies will form the j-th anti-optimal set of strategies $C_{s_2^j}^{-*}$.

$$\begin{aligned} C_{s_2^j}^{-*} : \{ s_1^i \in S_1 \mid \neg \exists s_1^{i'}, i' \in \{1, \ldots, I\} : \\ g_{i',j} \overset{min}{\succ} g_{i,j} \}, \forall i = 1, ..., I \end{aligned} \tag{12}$$

Mapping these anti-optimal strategies of P_2 into the objective space forms the j-th anti-optimal front: $F_{s_2^j}^{-*}$ where:

$$F_{s_2^j}^{-*} := \{ \bar{f}_{i,j} \in \Psi \mid s_1^i \in C_{s_2^j}^{-*} \} \tag{13}$$

All the anti-optimal fronts $F_{s_2^j}^{-*}$ of the introductory example are depicted in Figure 2, encircled by dashed curves. $F_{s_2^1}^{-*}$ includes the white diamond (4,1). $F_{s_2^2}^{-*}$ includes the gray triangle and square ((1,2) and (2,2) respectively). $F_{s_2^3}^{-*}$ includes the black square and diamond ((2,3) and (4,3) respectively), and $F_{s_1^4}^{-*}$ includes the dotted square (2,4), star (3,4) and diamond (4,4).

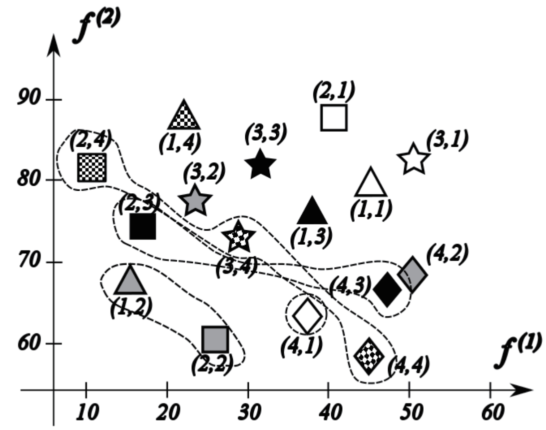

Fig. 2: All the anti-optimal fronts of the maximizer P_2

The next phase is to rank the strategies of P_2. In order to rank these strategies, the union of all of the anti-optimal sets of P_2 is constructed as follows:

$$SC_2 = \{C^{-*}_{s_2^1}, \ldots, C^{-*}_{s_2^j}, \ldots, C^{-*}_{s_2^J}\} \tag{14}$$

2) Assigning the rank grade in the worst-case approach: Next, sorting SC_2 and selecting the dominating sets, in the minimization problem (dominating in the worst-case sense), will result in the maximizer's (P_2) *irrational* strategies, which are the worst-case dominating strategies:

$$C_2^{w.c} := \{s_2^j \in S_2 \mid \neg\exists C^{-*}_{s_2^{j'}} \subset SC_2 : C^{-*}_{s_2^{j'}} \overset{max}{\succ} C^{-*}_{s_2^j}\} \tag{15}$$

The set $C_2^{w.c}$ includes all P_2's strategies that are associated with a dominating anti-optimal front in the minimization problem.

The set of rational strategies of P_2 is the relative complement of $C_2^{w.c}$ in SC_2:

$$C_2^R = SC_2 - C_2^{w.c} \tag{16}$$

The set C_2^R includes all P_2's strategies associated with a non-dominating anti-optimal front in the minimization problem.

Each of the maximizer rational strategies is represented in the objective space by its related anti-optimal front, which is termed as the maximizer rational front:

$$F^R_{s_2^j} := \{\bar{f}_{i,j} \in \Psi \mid s_1^i \in C^{-*}_{s_2^j} \wedge C^{-*}_{s_2^j} \subseteq C_2^R\} \tag{17}$$

All the strategies associated with the rational fronts are in the first rank ($Rank = 1$). In the given example, there are three strategies in the first rank, s_2^1, s_2^3 and s_2^4. In order to find the strategies in the next rank, all the strategies associated with $F^R_{s_2^j}$ are removed from the set of strategies SC_2. The procedure described in equations 15, 16 and 17 is repeated until all strategies are ranked.

TABLE I: The rank and in-rank grade of the subsets represented on figure 2

Strategy	s_2^1	s_2^2	s_2^3	s_2^4
Filling	white	gray	black	dotted
Rank	1	2	1	1
In-rank grade	73	-	69	81
Sorting	2	4	3	1

3) Assigning the in-rank grade: In the current problem each solution consists of a set of fronts in objective space (the associated anti-optimal representative front). The strategies with the same rank form a Pareto layer [7]. In [8] a measure of fitness (here termed as the "in-rank grade") within the same rank (layer) is introduced. The in-rank-grade allows promoting fronts that are more diverse, with smaller spread and are convex [8]. It is based on the distance a front needs to be shifted to become non-dominated by any other front. The procedure to obtain this measure and illustrative examples are detailed in [3]. The grades of the strategies of player P_2, of the example, are given in table I. Note that strategy s_2^2 has no grade because it is the only strategy in the second rank.

4) Sorting the strategies: The sorting of the strategies is a lexicographical process, which resembles the sorting process in NSGA-II [14]. First, the ranks are compared and the lower preferred. If the rank of two compared strategies is equal, the in-rank grade is compared and the strategy with the higher in-rank grade is preferred. The results of the sorting procedure are also shown on the third line in table I.

IV. Co-evolutionary algorithm

The MOG is solved for both players by a co-evolutionary algorithm with complete mixing. The evaluation of the strategies is done using the worst-case set-based sorting, as described in section III-B.

The pseudo-code of the proposed set-based co-evolutionary algorithm is described under Algorithm 1. It solves the MOG for *both* players simultaneously. At each generation, for each player (DM), the algorithm applies the player's strategy optimization procedure based on the worst-case set-based measures. The sorting procedure, which is described in section III-B4, is applied in the `elite` and in the `tournament` procedures (Algorithm 2 and Algorithm 3 respectively).

V. Example: Tug-of-war

The chosen example is a simple differential MOG, which is based on a SOG from [15]. A differential game is an interaction between two or more DMs *"wherein the evolution of the state is described by a differential equation and the players act throughout a time interval"* [15]. Extending the SOG, from [15] to a simple MOG, provides an opportunity to test the proposed algorithm. This is achieved since that the results can be compared with the physically expected results. This is in contrast to most MOGs, which are too complex and their results are hard to predict (e.g.,[16]).

Algorithm 1: Pseudo-code of the co-evolutionary algorithm

Input : Population size I and J of players P_1 and P_2, respectively; the number of generations N
Output : The elite populations E^1_{t+1}, E^2_{t+1}
Initialize: $E^1_{t+1} = \emptyset$, $t = 1$ and $E^2_{t+1} = \emptyset$

for $t = 1$ **to** N **do**
  Combine parent and elite populations to create two populations $R^1_t = Q^1_t \cup E^1_t$ and $R^2_t = Q^2_t \cup E^2_t$
  Perform all the $2I \times 2J$ interactions between all the strategies s^i_1 , $i = 1, ..., 2I$, of player P_1 and all the strategies s^j_2 , $j = 1, ..., 2J$, of player P_2
  For each strategy s^i_1 and s^j_2 assign the rank r_i and r_j and the in-rank grade f_i and f_j
  Select the elite populations E^1_{t+1} and E^2_{t+1} out of R^1_t and R^2_t respectively, based on the rank and the fitness by the `elite` procedure (see algorithm 2)
  Select the new offspring population O^1_{t+1} and O^2_{t+1} out of R^1_t and R^2_t respectively, based on the rank and the fitness by the `tournament` procedure (see algorithm 3)
  Perform `Cross-Over` procedure on the offspring populations O^1_{t+1} and O^2_{t+1}
  Perform `Mutation` procedure on the offspring populations O^1_{t+1} and O^2_{t+1}
  Set the new parent population $Q^1_{t+1} \leftarrow O^1_{t+1}$ and $Q^2_{t+1} \leftarrow O^2_{t+1}$
end

Algorithm 2: Pseudo-code of the `elite` procedure

```
/* This procedure is detailed for P1 only. For P2, replace index 1 by 2 and I,i by J,j, respectively. */
```

Input : The combined population R^1 of size $2I$
Output : The elite population E^1 of size I
Initialize: $k = 1$

Find all individuals in R^1 with rank $r = k$ to form population p_k
Set elite population $E^1 = \emptyset$
while *Size of* $\mid E^1 \mid + \mid p_k \mid \leq I$ **do**
  Combine elite population and p_k to form the new elite populations $E^1 \leftarrow E^1 \cup p_k$
  $k \leftarrow k + 1$
  Find all individuals in R^1 with rank $r = k$ to form population p_k
end
if $\mid E^1 \mid \leq I$ **then**
  Sort, from higher to lower, the population p_k by the individuals' in-rank grade f_i
  Add the $I- \mid E^1 \mid$ individuals of p_k to the elite $E^1 \leftarrow E^1 \cup p_k(1 \; to \; I- \mid E^1 \mid)$
end

Algorithm 3: Pseudo-code of the `tournament` procedure

```
/* This procedure is detailed for P1 only. For P2, replace index 1 by 2 and I,i by J,j, respectively. */
```

Input : The combined population R^1 of size $2I$
Output : The offspring population O^1 of size I
Initialize: Offspring population $O^1(1) = \emptyset$

for $i = 1$ **to** I **do**
  Select two individuals out of R^1 with random indexes $m, n \in [1, 2 \times I]$, ($R^1(n)$ and $R^1(m)$)
  if $r_n < r_m$ **then**
    $O^1(i+1) \leftarrow O^1(i) \cup R^1(n)$
  end
  else if $r_n > r_m$ **then**
    $O^1(i+1) \leftarrow O^1(i) \cup R^1(m)$
  end
  else if $r_n = r_m$ **then**
    if $f_n \geq f_m$ **then**
      $O^1(i+1) \leftarrow O^1(i) \cup R^1(n)$
    end
    else
      $O^1(i+1) \leftarrow O^1(i) \cup R^1(m)$
    end
  end
  $i \leftarrow i + 1$
end

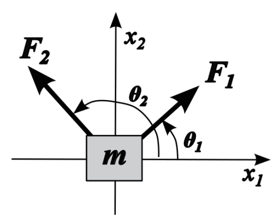

Fig. 3: The game setting of the tug-of-war MOG

A. The game setting

The setting includes a point mass m that can be moved on a frictionless horizontal plane as illustrated in figure 3.

The coordinates that describe the position of the object are x_1 and x_2. Player P_1 chooses to apply force F_1 at angle θ_1 and player P_2 chooses to apply force F_2 at angle θ_2, which are measured as seen in figure 3. In this game, choosing a strategy means choosing an angle. The game outcome (the payoff vector) will be the two coordinates of the position of the mass at the final time t_f. The objectives of player P_1 are to maximize the coordinates x_1 and x_2 while the objectives of player P_2 are to minimize those two same coordinates. Therefore, the optimization problems are:

$$for \;\; P_1/P_2 \;\; max/min(x_1, x_2) \tag{18}$$

As the two forces are of constant magnitude and constant direction, the acceleration of the mass is also constant. Its components along the two axes are: $\ddot{x}_1 = F_1cos(\theta_1) + F_2cos(\theta_2)$ and $\ddot{x}_2 = F_1sin(\theta_1) + F_2sin(\theta_2)$. The coordinates of the position of the mass are: $x_1(t)_1 = x_1(0) + \ddot{x}_1(0)t + \frac{1}{2}\ddot{x}_1t^2$ and $x_2(t) = x_2(0) + \ddot{x}_2(0)t + \frac{1}{2}\ddot{x}_2t^2$. To simplify the problem, the initial conditions are taken as: $x_1(0) = x_2(0) = 0$ and $\dot{x}_1(0) = \dot{x}_2(0) = 0$. Substituting the accelerations and the initial conditions, the coordinates of the mass at t_f are: $x_1(t_f) = x_1(\theta_1, \theta_2) = \frac{1}{2}\ddot{x}_1t_f^2 = (F_1cos(\theta_1) + F_2cos(\theta_2))\frac{1}{2}t_f^2$ and $x_2(t_f) = x_2(\theta_1, \theta_2) = \frac{1}{2}\ddot{x}_2t_f^2 = (F_1sin(\theta_1) + F_2sin(\theta_2))\frac{1}{2}t_f^2$. The payoff-vector of this MOG is $\bar{f}_{i,j} = [x_1(\theta_1^i, \theta_2^j), x_2(\theta_1^i, \theta_2^j)]$. The MinMax and MaxMin problems of this game, following equations 9 and 10, are:

$$\text{For the minimizer } P_1: \quad \min_{\theta_1^i \in \Theta_1} \max_{\theta_2^j \in \Theta_2} \bar{f}_{i,j} \tag{19}$$

$$\text{For the maximizer } P_2: \quad \max_{\theta_2^j \in \Theta_2} \min_{\theta_1^i \in \Theta_1} \bar{f}_{i,j} \tag{20}$$

The players solve these equations to find their rational strategies under the incomplete information, using worst-case optimization as discussed in section III-B.

B. Numerical details and results

The results are obtained using $F_1 = F_2 = 1N$, and $m = 1kg$. The final time is $t_f = \sqrt{2}sec$, which is chosen to provide a unit coefficient for $\ddot{x}$. The initial conditions are: $x_1(0) = x_2(0) = 0$ and $\dot{x}_1(0) = \dot{x}_2(0) = 0$. The design parameters (strategies) are the angles of the forces $0 \leq \theta_1 \leq 2\pi$ and $0 \leq \theta_2 \leq 2\pi$. Note that the number of possible strategies is infinite within this range. In the simulations, a uniform mutation with probability $p_m = 5\%$, and an arithmetical crossover with probability $p_c = 50\%$, were used. The population size is $I = 50$ for P_1 and $J = 50$ for P_2. The total number of generations used is 50. The problem is solved using the algorithm presented in Section IV.

Due to the simplicity of the considered MOG, repeated runs resulted in very similar results. In the current work we have not used any complicated problem; for the given problem a statistical data of the runs appears to be of a minor relevance (see future work in section VI).

Figures 4, 5, 6, and 7 show the obtained results after one, five, ten and fifty generations respectively. At each figure the top panel displays the objective space and the bottom panel displays the design (strategies) space. Each point at the top panels marks the position of the mass as a result of one interaction between the two players. The resulting positions due to the interactions between the rational strategies of both players are marked by black points and the rest are marked by gray points. The bottom panels of these figures show the strategies as obtained by the algorithm at the specified generation. The strategies of the minimizer and the maximizer are denoted by circles and crosses respectively. The rational strategies are marked by black circles and crosses, whereas the rest (irrational) are marked by gray color.

It is clear from figures 4, 5, 6 and 7 that for the given problem convergence is fast. These numerical results have

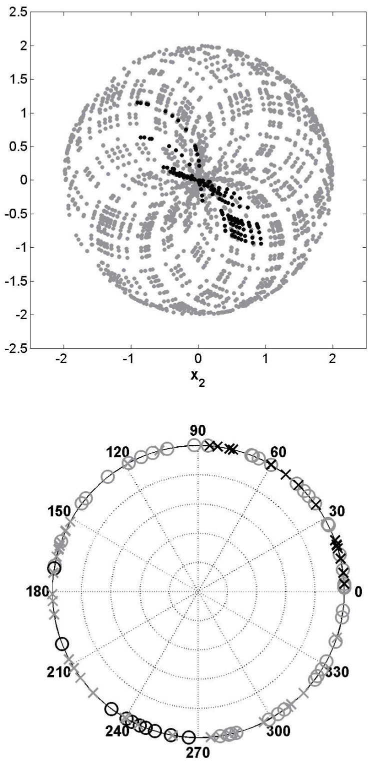

Fig. 4: The performances after one generations

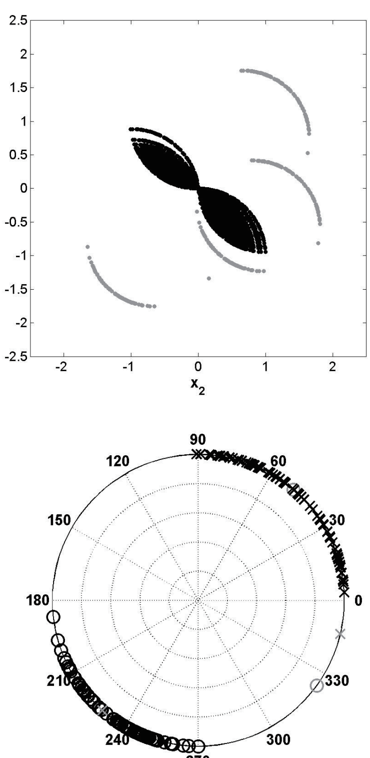

Fig. 5: The performances after five generations

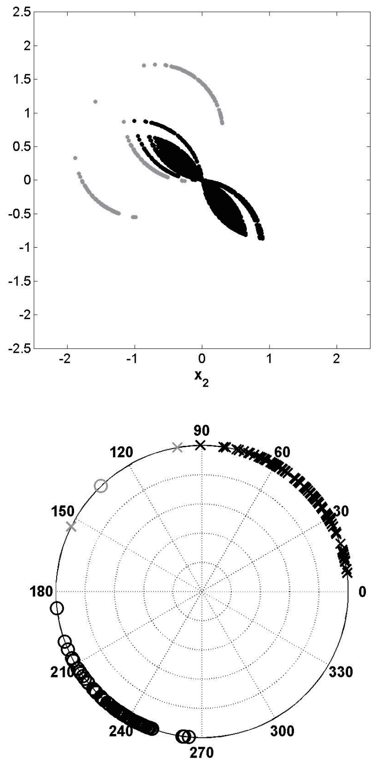

Fig. 6: The performances after ten generations

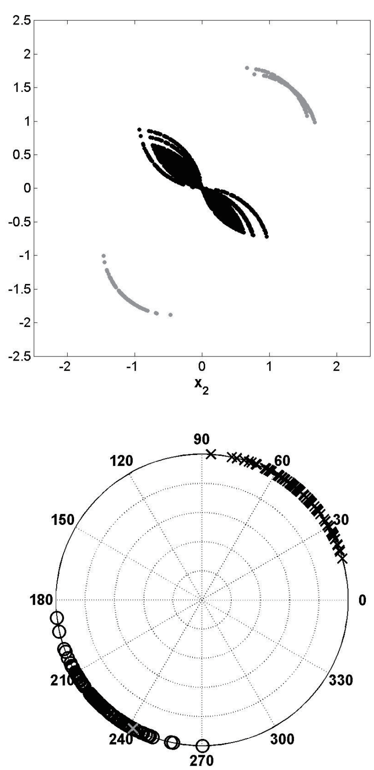

Fig. 7: The performances after fifty generations

a clear physical interpretation, which is described in the following subsection.

C. Physical interpretation of the rational strategies

The obtained rational strategies can be justified using physical arguments. Since that the minimizer wants the mass to be at the third quadrant, it has to pull in a direction within that quadrant, and vice versa for the maximizer. Moreover, due to the postponed preference, any force within that range of directions is valid, and has a corresponding case of preference that it will be the optimal. On the other hand if the minimizer, for example, chooses to pull in a direction within the second quadrant, it means that it will lose performance in the second payoff. In summary, the rational strategies for the maximizer are in the range of $0 \leq \theta_1 \leq \frac{1}{2}\pi$, and those of the minimizer are in the range $\pi \leq \theta_2 \leq \frac{3}{2}\pi$. Indeed, these are the angles obtained by the co-evolutionary algorithm as seen in figure 6.

VI. Summary, conclusions and future work

In contrast to most studies on MOGs, which use a utility approach to handle the multiplicity of objectives, here objective preferences are postponed by both players. As neither player knows the opponent's preferences, each player adopts a worst-case approach. To solve such games, a set-based co-evolutionary algorithm is proposed. In contrast to the procedure in [11], the suggested algorithm finds the solution for both players simultaneously.

A simple differential MOG (tug-of-war) is used to illustrate the method and demonstrates the applicability of the suggested algorithm. The obtained numerical results are in agreement with physical argumentations. While demonstrating the validity of the proposed co-evolutionary algorithm, on a problem with infinite strategic space, the suggested code and MOG approach require further studies.

In future work we intend to employ the proposed algorithm on more complex differential MOGs such as multi-objective interception game (e.g., [13]), and cyber security games (e.g., [16]). Due to concerns about the red queen effect and the expected problem of intransitivities, the current algorithm will be compared with the one in [3]. It is noted that customary test functions and measures are commonly used to evaluate the characteristics of multi-objective optimization algorithms (e.g., [6]). However, for MOGs, there is a need to formulate such functions and measures to evaluate the algorithms. This will be a part of the intended future work.

Acknowledgment

This research was supported by a Marie Curie International Research Staff Exchange Scheme Fellowship within the 7^{th} European Community Framework Programme. The first and second authors would like to acknowledge the hospitality and support of the Mechanical and Material Engineering Department at the University of Western Ontario, Canada.

References

[1] M. Zeleny, “Games with multiple payoffs,” *International Journal of Game Theory*, vol. 4, no. 4, pp. 179–191, 1975.

[2] H. W. Corley, “Games with vector payoffs,” *Journal of Optimization Theory and Applications*, vol. 47, no. 4, pp. 491–498, 1985. [Online]. Available: http://dx.doi.org/10.1007/BF00942194

[3] G. Avigad, E. Eisenstadt, and M. Weiss-Cohen, “Optimal strategies for multi objective games and their search by evolutionary multi objective optimization,” in *Computational Intelligence and Games (CIG), 2011 IEEE Conference on*, Mechanical Engineering Department at Ort Braude College of Engineering, Karmiel, Israel. IEEE, 2011, pp. 166–173.

[4] E. Eisenstadt, G. Avigad, A. Moshaiov, and J. Branke, “Multi-Objective Zero-Sum Games with Postponed Objective Preferences,” Tech. Rep., 2014. [Online]. Available: http://www.eng.tau.ac.il/ moshaiov/MOG_8.pdf

[5] D. A. Van Veldhuizen and G. B. Lamont, “Multiobjective evolutionary algorithms: analyzing the state-of-the-art.” *Evolutionary computation*, vol. 8, no. 2, pp. 125–47, Jan. 2000.

[6] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. Da Fonseca, “Performance assessment of multiobjective optimizers: An analysis and review,” *Evolutionary Computation, IEEE Transactions on*, vol. 7, no. 2, pp. 117–132, 2003.

[7] G. Avigad, E. Eisenstadt, and A. Goldvard, “Pareto layer: Its formulation and search by way of evolutionary multi-objective optimization,” *Engineering Optimization*, vol. 42, no. 5, pp. 453–470, Feb. 2010. [Online]. Available: http://dx.doi.org/10.1080/03052150903271959

[8] G. Avigad and J. Branke, “Embedded evolutionary multi-objective optimization for worst case robustness,” in *Proceedings of the 10th annual conference on Genetic and evolutionary computation*. ACM, 2008, pp. 617–624.

[9] W. D. Cook, “Zero-sum games with multiple goals,” *Naval Research Logistics Quarterly*, vol. 23, no. 4, pp. 615–621, Dec. 1976.

[10] E. L. Hannan, “Reformulating zero-sum games with multiple goals,” *Naval Research Logistics Quarterly*, vol. 29, no. 1, pp. 113–118, Mar. 1982.

[11] G. Avigad, E. Eisenstadt, and V. Glizer, “Solution of Multi-objective Min-Max and Max-Min Games by Evolution,” in *Evolutionary Multi-Criterion Optimization SE - 21*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2013, vol. 7811, pp. 246–260.

[12] ——, “Evolving a Pareto Front for an Optimal Bi-objective Robust Interception Problem with Imperfect Information,” in *EVOLVE - A Bridge between Probability, Set Oriented Numerics, and Evolutionary Computation II SE - 8*, ser. Advances in Intelligent Systems and Computing. Springer Berlin Heidelberg, 2013, vol. 175, pp. 121–135.

[13] D. Ghose, U. R. Prasad, and C. Math, “Multicriterion differential games with applications to combat problems,” *Computers & Mathematics with Applications*, vol. 18, no. 13, pp. 117–126, 1989.

[14] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, “A fast and elitist multiobjective genetic algorithm: NSGA-II,” *Evolutionary Computation, IEEE Transactions on*, vol. 6, no. 2, pp. 182–197, 2002.

[15] T. Basar and G. J. Olsder, *Dynamic noncooperative game theory*. SIAM, 1995, vol. 200.

[16] V. Viduto, W. Huang, and C. Maple, “Toward Optimal Multi-Objective Models of Network Security: Survey,” *Automation and Computing (ICAC), 2011 17th International Conference on*, no. September, pp. 6–11, 2011.