

Optimal Strategies for Multi Objective Games and Their Search by Evolutionary Multi Objective Optimization

G. Avigad, E. Eisenstadt, M. Weiss Cohen

Abstract. While both games and Multi-Objective Optimization (MOO) have been studied extensively in the literature, Multi-Objective Games (MOGs) have received less research attention. Existing studies deal mainly with mathematical formulations of the optimum. However, a definition and search for the representation of the optimal set, in the multi objective space, has not been attended. More specifically, a Pareto front for MOGs has not been defined or searched for in a concise way. In this paper we define such a front and propose a set-based multi-objective evolutionary algorithm to search for it. The resulting front, which is shown to be a layer rather than a clear-cut front, may support players in making strategic decisions during MOGs. Two examples are used to demonstrate the applicability of the algorithm. The results show that artificial intelligence may help solve complicated MOGs, thus highlighting a new and exciting research direction.

I. INTRODUCTION

In conventional game theory problems, the usual assumption is that decision makers (players) make their decisions based on a scalar payoff. But in many practical problems in economics and engineering, decision makers must cope with multiple objectives or payoffs. In such problems, a vector of objective functions must be considered. A MOG may be defined as a problem in which one player aims to minimize a set of objectives, while the other player aims at maximizing these objectives. As an example of such a game, consider a player who possesses a set of boats. These boats are to set sail from several possible docking locations, so that one of them will reach a target point in the shortest time possible. While sailing, the boats must escape an opponent vessel that is guarding the target and is capable of traveling twice as fast as the boats. When the vessel intercepts a boat, that boat is eliminated from the game. The MOG is defined such that the first opponent seeks to minimize the distance traveled by one of its boats (minimizing the time taken to get to the target) while also minimizing its loss of boats (sacrificed for the sake of optimization). On the other hand, the opponent's objectives are to maximize the number of intercepted boats and to maximize the time (distance) taken for any of the

opponent's boats to reach the target point. The strategies adopted by the first player may include, but are not limited to, where the boats should start from and how many and which boats should be sacrificed. The strategies used by the vessel may include where to start, when to start and whether to intercept the boat closest to the target or the nearest boat.

As in the case of single objective games, the notion of equilibrium can be defined in terms of unfruitful deviation, from the equilibrium strategies. According to [1], in the MOG setting this can be interpreted as follows:

Deviations from the equilibrium strategies do not offer any gains to any of the pay-off functions for any of the players.

All existing studies related to MOGs refer to definitions for this equilibrium (e.g., [2]). The predominant definition of this equilibrium, termed the Pareto-Nash equilibrium, has been suggested in [3]. The Pareto-Nash equilibrium uses the concept of cooperative games, because according to this notion, sub-players under the same coalitions should, according to the Pareto notion, optimize their vector functions on a set of strategies. On the other hand, this notion also takes into account the concept of non-cooperative games, because coalitions interact on the set of situations and are interested in preserving the Nash equilibrium between coalitions. An extension to the work in [3] may be found in [1], where several algorithms have been suggested for searching for these equilibrium strategies.

MOGs are commonly solved by linear programming, e.g. [4], in which a multiple objective zero-sum game is solved. A non-zero-sum version of a MOG has been solved by Contini [5]. Detailed algorithms for finding Pareto Nash equilibrium-related strategies may be found in [1] and [3]. In all of these cases, weights are altered in order to search for one equilibrium solution at a time. This means the algorithms must be executed sequentially in order to reveal more equilibrium points. Artificial intelligence-based approaches have been applied to MOGs within the framework of "fuzzy multi-objective games." In such studies (e.g., [6]) the objectives are aggregated to a surrogate objective in which the weights (describing the players' preferences towards the objectives) are modeled through fuzzy functions.

II. BACKGROUND

A. MOO and EMO

The topic of MOO concerns the search for solutions to many real world problems, dubbed appropriately enough Multi-objective Problems (MOPs). In cases of contradicting objectives, there is no universally accepted definition of an

A. Avigad is with the Mechanical Engineering Department at Ort Braude College of Engineering, Karmiel, Israel (phone: 97249901767; fax: 97249901866; e-mail: gideon@braude.ac.il).

E. M. Eisenstadt is with the Mechanical Engineering Department at Ort Braude College of Engineering, Karmiel, Israel (e-mail: erez@braude.co.il).

M. C. Weiss is with the Computer Science Department at Ort Braude College of Engineering, Karmiel, Israel (e-mail: miri@braude.co.il).

'optimum' as in a single-objective optimization [7]. In such cases, there is also no single global solution, thus often rendering it useful to determine a set of solutions that fits a predetermined definition for an optimum and let the decision maker choose among them. The predominant concept in defining such a set of points is Pareto optimality.

The classical multi-objective search problem, such as the one dealt with in [8], is to simultaneously minimize all the objective functions $[f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_k(\mathbf{x})]^T$ subject to $\mathbf{x} = [x_1, x_2, \dots, x_n]^T \in \Omega \subseteq R^n$. For the sake of brevity, this is formulated as Minimize $\mathbf{F}(\mathbf{x})$, where

$\mathbf{F}(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_k(\mathbf{x})]^T$ and Ω is the feasible design space. Since in most cases the objectives are conflicting, the relation of domination between decision vectors is important. The concept of domination is that a vector $\mathbf{x} \in \Omega$ is said to dominate a vector $\mathbf{y} \in \Omega$, (in short, $\mathbf{x} < \mathbf{y}$), if $f_i(\mathbf{x}) \leq f_i(\mathbf{y})$ for all $i = 1, \dots, k$ and there exists $j \in \{1, \dots, k\}$ with $f_j(\mathbf{x}) < f_j(\mathbf{y})$. A solution $\mathbf{x}^* \in \Omega$ is called a Pareto optimal if no solution exists in Ω that dominates \mathbf{x}^* . The set of all optimal solutions is the Pareto set $P := \{\mathbf{x} \in \Omega : \neg \exists \mathbf{y} \in \Omega : \mathbf{y} < \mathbf{x}\}$. Its image, $F(P)$, is the Pareto front.

Searching a multi-objective design space for optimal solutions (and for their related front) by evolutionary computation approaches is commonly referred to as Evolutionary Multi-Objective Optimization (EMO). A Multi-Objective Evolutionary Algorithm (MOEA) is an EMO algorithm that searches for a solution in a multi-criteria space based upon some insights from evolutionary theories. Detailed descriptions of multi-objective evolutionary techniques can be found in [8].

B. Set-based Design and Evolution

Sets have also been compared within the framework of evolutionary multi-objective optimization. For instance, some studies use set domination to search for the best approximation of the Pareto front (e.g., the indicator-based evolutionary algorithm-IBEA [9]). Such a search is based on assigning a value to the degree of domination between sets of competing approximations of the Pareto set. For example, such assignment is carried out using the binary additive indicator, introduced in [10]. Considering more than two sets for the purpose of evolving a front has also been examined. The indicators used to compare two sets (e.g., the binary additive indicator) may serve as a measure for comparing between all the sets. Among such suggested comparisons are the two fitness assignments considered in conjunction with the IBEA algorithm. An updated review and recent developments on evolving sets of solutions for evolving an approximation for the Pareto front can be found in [11].

Comparison of sets is not limited to the search for good Pareto approximations. In these works, each solution of the MOP is associated with multiple performance vectors within

the objective space. A decision maker must choose between solutions by considering multiple representations for each solution. Examples include studies of the Pareto layer [12] and the worst case EMO [13]. Of special interest to the current paper is the study by Avigad and Branke [13], in which a solution is evaluated by means of a finite set of different worst scenarios. It was assumed that a solution cannot be reduced to a single worst-case representative because different users would consider different representatives as their worst case. This assumption led to the need to compare between solutions based on sets of worst cases. For this purpose, a definition of dominance for worst-case optimization based on sets of representatives has been suggested. Comparing among non-dominated solutions has been also suggested. The worst case EMO bears some similarity to the current case. However, when dealing with uncertainty (as is the case in [13]), approaches other than the worst case approach might be considered, for example the use of statistical data, whereas here the use of statistics is not an option.

C. Shortcomings of current state of the art

In current research studies examining MOGs, optimal equilibrium points are defined. This means that for one opponent's optimal strategy, an optimal strategy is also defined for the other opponent, so that equilibrium is gained. Although we do not question these definitions, we believe that the search for a set of optimal strategies has several major shortcomings:

1. We envisage that many practical MOGs would be very hard to solve using non-stochastic approaches. The linear programming used currently would not yield good results for non-linear mapping between the strategies space and the objective space. Moreover, the non-continuity and concavity of the image sets would prevent common gradient-based approaches from enhancing a productive search. Furthermore, as stated in [3], discrete games cannot be solved directly and are currently mapped into continuous games by introducing a combination of continuous functions.
2. Presentation of all the equilibrium points would not necessarily support players in making decisions while playing MOGs. Finding an equilibrium point means that strategies for the players are found. However, for multi-objective spaces it is possible that one opponent's strategy may be countered by several optimal strategies taken by the other opponent. Currently, finding these optimal strategies is a matter of chance and nothing more. The above conviction is also related to the fact that representation of the set of all Pareto Nash equilibrium replies, in the objective space has not been defined or characterized.
3. Although EMO seems to be a major approach, and in some cases the only possible approach, to search for optimal solutions for MOPs, it has not been applied to MOGs (Multi objective optimization has been used with

the context of games yet mostly for evolving strategies (e.g. [14]) and not for MOGs)

4. Although learning is a major issue in games, it has not been utilized within the context of MOGs.

In the current paper, we aim to advance the state of the art to pave the way for treating these shortcomings. Specifically, we propose a definition for presentation of the optimal set in the objective space. This definition is related to how players should be supported while playing MOGs. Moreover, a MOEA is proposed for searching this representation. We expect that EMO will contribute to solving MOGs, as it has to MOPs. Learning how to play MOGs based on the knowledge acquired is a subject of future research.

III. METHODOLOGY

A. The approach

Verbal descriptions and schematic drawings are used to clarify the approach put forward in this paper. The next subsection provides mathematical definitions and formulations. Considering the boats and vessel example in the introduction, suppose the strategy for the vessel is determined (e.g., the vessel intercepts the boat closest to the target). For such a strategy, the opponent (in charge of the boats) may adopt a number of different strategies (e.g., sacrifice one boat or two boats, pick a mutual starting point for all boats or spread them out). This means that for one chosen strategy chosen by one opponent, there may be a set of games, each involving this strategy, and a strategy (one of several) chosen by the other opponent. The results of the game may be evaluated through objective functions (in this case, the distance traveled by the boat that reached the target or was closest to it, if intercepted, and the number of intercepted boats). Therefore, in the current example each game may be represented by a vector of two components in the objective space (number of lost boats and distance). As a result, each strategy chosen for the vessel may be represented in the objective space by a cluster of points, each associated with one possible game. The question is to determine which of the strategies adopted by the boats, are optimal strategies. If a common multi-objective optimization is applied to this problem, a Pareto front associated with this specific vessel's strategy will be found (see [15]). This Pareto front represents the best strategies that may be adopted by the boats so as to counteract the strategy adopted by the vessel. Consequently, it is conceivable that any strategy adopted by the vessel would involve a Pareto front representing the optimal strategies utilized by the boats to counter it. The next stage is now to optimize so that these fronts are worst with respect to the boats. This means that the sets (Pareto fronts) are utilized to find the best strategies that should be taken by the vessel, although countered by the best strategies taken by the boats, as illustrated in Figure 1. The left panel of the figure depicts six strategies (designated by different colored symbols) for the vessel. Each strategy has a set of strategies adopted by the boats. Therefore there are six clusters in the bi-objective

space. For the sake of simplicity, suppose these are the only possible strategies (for both the vessel and the boats). For each cluster there is a related front, the result of minimizing the two objectives. The fronts are depicted in the right panel of the figure. The optimal strategies for the vessel are now found by maximizing the objectives. The result is a set of strategies, which are represented by their related fronts and circled in the figure.

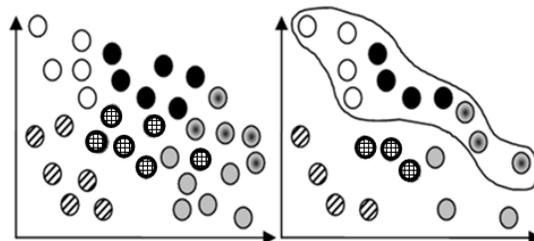


Figure 1: Left panel: Six clusters of performances, each related to a different strategy of one of the opponents; Right panel: The Pareto fronts of each cluster and the optimal set of related fronts (circled).

The following should be noted: a) The optimal strategies for the vessel may form a layer of representations in the objective space and not necessarily a clear-cut front (as is a Pareto front); b) A strategy for the vessel should be chosen out of the optimal set (circled); otherwise the boats may choose strategies that will result in better performances (for the boats) when minimization is considered; c) Choosing optimal strategies for the boats should be based on another optimization problem starting with clusters of strategies for the boats; d) Finding the Pareto set for each cluster is an optimization problem, solved here through an embedded EMO algorithm; e) Finding the Pareto layer calls for comparing sets (the Pareto fronts of each strategy). Here this search is conducted through set-based EMO.

B. Problem Definition

A strategy is searched for opponent 1 such that it will have optimal performances in a multi-objective space under the best performances of opponent 2, who chooses another strategy (and starts the game). Let the two opponents be associated with a set of possible strategies as follows:

$$S^{(1)} = \{s_1^{(1)}, \dots, s_i^{(1)}, \dots, s_n^{(1)}\}, S^{(2)} = \{s_1^{(2)}, \dots, s_j^{(2)}, \dots, s_m^{(2)}\}$$

$$\text{where } S^{(1)} \in \Omega \subseteq R^n, S^{(2)} \in \Gamma \subseteq R^m.$$

$s_i^{(1)}$ and $s_j^{(2)}$ are the i -th and j -th strategies of opponent 1 and 2, respectively, out of n and m possible strategies for these opponents, respectively. A game is a result of the opponents each choosing a strategy. A game between opponent 1 playing with strategy $s_i^{(1)}$ and opponent 2 playing with strategy $s_j^{(2)}$ is $g_{i,j}$, such that: $g_{i,j} \subseteq \Phi = \Omega \times \Gamma$. The optimization problem may be formulated as a MOP:

$$\max_{s_j^{(2)}} \min_{s_i^{(1)}} F(g_{i,j}) \quad (1)$$

where $F \rightarrow \Phi: \Psi \subseteq R^K$. K is the number of MOP objectives, such that:

$$F(g_{i,j}) = [f_1(g_{i,j}), \dots, f_K(g_{i,j})]^T \quad (2)$$

C. Problem solution

For each strategy adopted by opponent 2 (say the j^* -th strategy), there are n possible strategies for opponent 1. All such possible games form a set: $G_{j^*} \subseteq \Phi = [g_{1,j^*}, g_{2,j^*}, \dots, g_{n,j^*}]^T$. Optimizing the strategies of opponent 1 with respect to the j^* -th strategy means that the following optimization problem should be solved:

$$\min_{s_i^{(1)}} F(g_{i,j^*}) \quad (3)$$

Clearly, if the objectives are contradicting there may be a set of opponent 1 strategies that will serve as solutions to this problem. These will form a Pareto set:

$$G^*_{j^*} := \{g_{i,j^*} \in \Phi \mid \neg \exists g_{u,j^*} \in \Phi: g_{u,j^*} \prec g_{i,j^*}\} \quad (4)$$

The mapping of these optimal strategies to the objective space forms a Pareto front F^*_{i,j^*} , which is associated with the j^* -th strategy (adopted by opponent 2). Equation 1 may now be realized by searching for a strategy in order to:

$$\max_{\substack{s_j^{(2)} \\ g_{i,j^*} \in G^*_{j^*}}} F(g_{i,j^*}) \quad (5)$$

This means that the best strategy for opponent 2 is searched for while considering the best strategies adopted by opponent 1. This optimization results in an optimal set of strategies for opponent 2:

$$P^* := \{g_{i,j^*} \in G^*_{j^*} \mid \neg \exists g_{u,j^{**}} \in G^*_{j^{**}}: g_{u,j^{**}} \succ g_{i,j^*}\} \quad (6)$$

for all $j^* \neq j^{**}$

Please note that here \succ is used instead of \prec . P^* includes a set of strategies for opponent 2 that are each represented in the objective space by the best strategies adopted by opponent 1. Therefore, the mapping of the entire optimal set to the objective space, $F(P^*)$, results in a set of sets that forms a Pareto Layer (PL). Figure 2 highlights the different notions associated with the objective space.

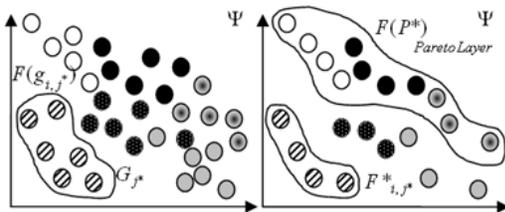


Figure 2: Left panel: six clusters of Figure 1 designating the

performances of one game $F(g_{i,j^*})$ and one of the clusters (G_{j^*}). Right panel: The Pareto front of each cluster (F^*_{i,j^*}) and the Pareto layer $F(P^*)$, circled.

D. Supporting decisions based on the Pareto Layer

A possible PL is depicted in Figure 3. The PL may support opponent 2 in making decisions (the opponent associated with the maximization). Based on the resulting PL, the following may be concluded: a) Choosing PL-related strategies for opponent 2 confines the performances of opponent 1 to be outside the light grey area; b) If opponent 2 prefers objective 2 over objective 1, he/she may choose the strategy involving the strip-related set. Opponent 1 will be then restricted to the dark grey area, which is associated with relatively high values for objective 2. If opponent 2 prefers objective 1 over objective 2, he/she will probably choose the black-grey related strategy; c) Choosing the white-related strategy might give opponent 1 too much freedom in choosing a counter strategy.

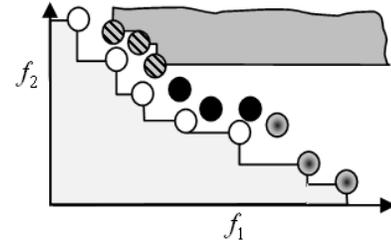


Figure 3: A PL and two designated areas. The bright grey area designates a region in the objective space that is unattainable by opponent 1 if opponent 2 chooses to use PL strategies. The dark region designates the performances to which opponent 1 will be confined if the strip-related strategy is chosen by opponent 2.

If opponent 1 is to be given support, the PL should be found by solving:

$$\min_{s_j^{(1)}} \max_{s_i^{(2)}} F(g_{i,j}) \quad (7)$$

According to [16], the two PLs will not be identical because the related equilibrium points will not be the same.

E. The evolutionary search

The evolutionary search is inspired by the evolutionary algorithms, represented in [12] and [13]. Based on these works, an embedded MOEA is employed here to find the Pareto set for each candidate strategy for opponent 2 (see Equation 3). Then the main algorithm is used to enhance a search pressure towards optimal strategies for opponent 2 (Equation 5) by set-based evolution.

To enhance the evolution, fitness is assigned to each individual. Fitness assignment is based on the rank to which the individual belongs and according to a set-based measure. The highest rank is assigned to strategies that are not at all dominated.

1. Ranking

L=1

While not all the individuals in the population are assigned a rank

1. Combine all the Pareto sets of the population's candidate strategies: $G_j = \bigcup G_j^*$
2. Find the Pareto front F_j of G_j (for a max-max problem).
3. If $\exists F(g_{i,j^*}) \in F_j$ then $G_{j^*} \in r_L$.
4. Remove all individuals of the L-th rank from the population.
5. L=L+1

End

2. Fitness assignment

Now suppose that the strategies do not dominate each other. In such a case, the j^* strategy is compared to the j^{**} strategy by using a set-based measure. We propose using the measure $I_{\phi^+}(j^*, j^{**})$, which is the minimum distance by which the Pareto front of the j^{**} -th strategy must be moved in either direction such that the assigned strategy (j^* -th strategy) front is strictly dominated by it (considering a max-max problem).

$$I_{\phi^+}(G_{j^*}^*, G_{j^{**}}^*) = \max_{g_{i,j^{**}} \in G_{j^{**}}^*} \min_{g_{i,j^*} \in G_{j^*}^*} \max_{1 \leq k \leq K} (f_k(g_{i,j^*}) - f_k(g_{i,j^{**}})) \quad (8)$$

And the overall fitness is:

$$Fit_{\phi}(G_{j^*}^*) = \min_{\forall G_{j^*}^* \in r_L} \{I_{\phi^+}(G_{j^*}^*, G_{j^{**}}^*)\} \quad (9)$$

Figure 4 clarifies the result obtained from using the suggested measure and ranking.

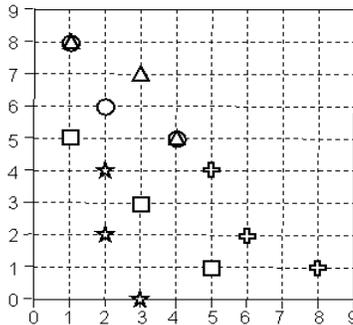


Figure 4: Five strategy-related Pareto fronts

The figure depicts five strategy-related Pareto fronts in a bi-objective space. The circle-, triangle- and plus-related strategies share the first rank (each has at least one non-dominated point when max-max is considered). The star- and square-related strategies share the second rank. Comparison of the circle-related set to the triangle-related set by means of

Equation 8 is detailed in Table 1.

Table 1: Set-based measure computation (Equation 8), comparing circle- and triangle-related strategies.

$f_k(g_{i,j^*}) = \bigcirc$ $f_k(g_{i,j^{**}}) = \triangle$				$f_k(g_{i,j^*}) = \triangle$ $f_k(g_{i,j^{**}}) = \bigcirc$			
$f_k(g_{i,j^*}) - f_k(g_{i,j^{**}})$	max	min	max	$f_k(g_{i,j^*}) - f_k(g_{i,j^{**}})$	max	min	max
(0,0)	0			(0,0)	0		
(-2,1)	1	0		(-1,2)	2	0	
(-3,3)	3			(-3,3)	3		
(1,-2)	1			(2,-1)	2		
(-1,-1)	-1	-1	0	(1,1)	1	1	1
(-2,1)	1			(-1,2)	2		
(3,-3)	3			(3,-3)	3		
(1,-2)	1	0		(2,-1)	2	0	
(0,0)	0			(0,0)	0		

All the pair wise comparisons yield:

$$I_{\phi^+}(G_{square}^*, G_{star}^*) = 2, I_{\phi^+}(G_{star}^*, G_{square}^*) = 1,$$

$$I_{\phi^+}(G_{circle}^*, G_{triangle}^*) = 0, I_{\phi^+}(G_{triangle}^*, G_{circle}^*) = 1,$$

$$I_{\phi^+}(G_{circle}^*, G_{plus}^*) = 4, I_{\phi^+}(G_{plus}^*, G_{circle}^*) = 4,$$

$$I_{\phi^+}(G_{triangle}^*, G_{plus}^*) = 4, I_{\phi^+}(G_{plus}^*, G_{triangle}^*) = 4$$

Based on Equation 9, the overall fitness of strategies belonging to Rank 1 is:

$$\begin{aligned} Fit_{\phi}(circle) &= Fit_{\phi}(I_{\phi^+}(G_{circle}^*, G_{plus}^*), I_{\phi^+}(G_{circle}^*, G_{triangle}^*)) \\ &= \min_{x^z \in P \setminus x} \{0, 4\} = 0 \end{aligned}$$

$$\begin{aligned} Fit_{\phi}(plus) &= Fit_{\phi}(I_{\phi^+}(G_{plus}^*, G_{circle}^*), I_{\phi^+}(G_{plus}^*, G_{triangle}^*)) \\ &= \min_{x^z \in P \setminus x} \{4, 4\} = 4 \end{aligned}$$

$$\begin{aligned} Fit_{\phi}(triangle) &= Fit_{\phi}(I_{\phi^+}(G_{triangle}^*, G_{plus}^*), I_{\phi^+}(G_{triangle}^*, G_{circle}^*)) \\ &= \min_{x^z \in P \setminus x} \{4, 1\} = 1 \end{aligned}$$

Based on Equation 9, the overall fitness of strategies belonging to Rank 2 is:

$$Fit_{\phi}(square) = Fit_{\phi}(I_{\phi^+}(G_{square}^*, G_{star}^*)) = 2;$$

$$Fit_{\phi}(star) = Fit_{\phi}(I_{\phi^+}(G_{star}^*, G_{square}^*)) = 1$$

As in [13], the fitness measure prefers fronts that are not crowded (compare the fitness for the circles, which are crowded, and for the pluses, which are not crowded) and convex (compare the fitness of the triangles and circles). These allow the evolution of strategies with dominating performances and the spread of the strategies along the PL (serves as a set-based crowding operator).

3. Comparison of two strategies

Based on III.e.1 and III.e.2, strategy j^* is preferred over strategy j^{**} if:

- a. $rank(G_{j^*}^*) < rank(G_{j^{**}}^*)$
- b. $rank(G_{j^*}^*) = rank(G_{j^{**}}^*) \wedge Fit_{\phi}(G_{j^*}^*) > Fit_{\phi}(G_{j^{**}}^*)$

The evolutionary multi-objective optimization algorithm

used to enhance a search pressure towards the PL is given by the following algorithm.

4. Algorithm

The algorithm that enhances a search pressure towards finding optimal strategies for a MOG is:

- a. Initialize a population P_t of size $n_1 = |P_t|$ which decodes the m opponent 2 strategies.
- b. Combine parent and offspring populations and create $R_t = P_t \cup Q_t$.
- c. For all individuals of R_t :
 - c.1 Initialize a population G_t of size $n_2 = |G_t|$ which decodes the n possible strategies for opponent 1.
 - c.2 Run NSGA-II [17] on the min-min optimization problem to find a related front for each individual of R_t (see equation 4).
- end
- d. Perform set-based domination sorting for R_t (max-max problem) and find ranks, $r_i, i=1, \dots, n_r$ where n_r is the number of ranks in a generation (see Section III.E.1).
- e. For $i=1$ to n_r
 - e.1 Compute fitness for all individuals of the same rank (See Equations 8 and 9).
 - e.2 Order the individuals of the rank into a list according to their fitness ($I_{fit}, >$).
- End
- f. Initialize a new parent population $P_{t+1} = \emptyset$.
- e. Include the first m individuals of I_{Fit} in the new parent population: P_{t+1} , to form an elite population.
- f. Create a population Q_{t+1}^* from P_{t+1} by tournament selection (see Section III.E.3).
- g. Perform crossover on Q_{t+1}^* to obtain Q_{t+1}^{**} .
- h. Perform mutation to obtain Q_{t+1} .
- i. If the last generation has not been arrived at, go-to 'b'.
- j. Introduce the PL (the Pareto fronts of all strategies which are included in P_{t+1} and belong to r_1).

IV. EXAMPLES

In this section, two examples are given. The first is an artificial example based on pure mathematical objective functions. The second is the vessel-boats example.

a. Artificial example

In this section, a common MOP is posed as a MOG. A bi-objective problem with the following objectives is considered:

$$f_1 = x_1^2 + x_2^2 + y_1^2$$

$$f_2 = \frac{y_2}{x_1 + x_2} + 10y_2$$

where $20 \leq x_1, x_2 \leq 20$; $0.5 \leq y_1, y_2 \leq 4$. In the current artificial example, it is assumed that opponent 1 chooses

strategies that are merely the values for parameters x_1 and x_2 and aims at minimizing the two objectives. Opponent 2 maximizes these objectives by searching for optimal strategies (values) for y_1 and y_2 . Figure 5 depicts one individual from the first generation. The figure depicts both the initial population (step c1 of the algorithm) for this individual (blank circles) and its evolved Pareto front (step c2 of the algorithm), designated by x signs.

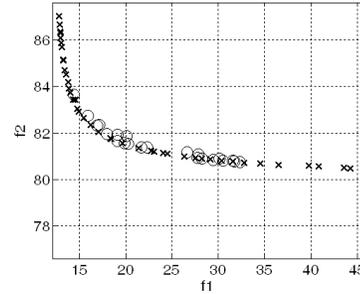


Figure 5: Initial inner population of a strategy (blank circles) and the evolved Pareto front (x signs), showing the evolution of non-dominated solutions scattered along the front.

Figure 6 depicts the initial elite population (step 'e' of the algorithm) in blue and the evolved PL in red. Clearly, the set-based EMO has pushed the Pareto fronts towards maximization of both objectives, as well as spread them along the PL. Figure 7 depicts the PL with the different strategies designated by different colors, highlighting the layer characteristic of the PL. Based on the PL, the opponent that chooses y_1 and y_2 may choose them so as to ensure maximization of the objectives under the inspiration of the opponent, who chooses x_1 and x_2 for minimization.

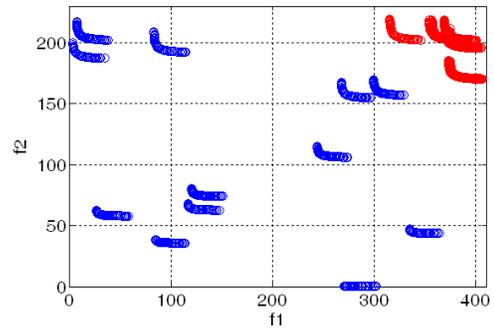


Figure 6: Initial population-related set of Pareto fronts (blue) and the evolved PL (red).

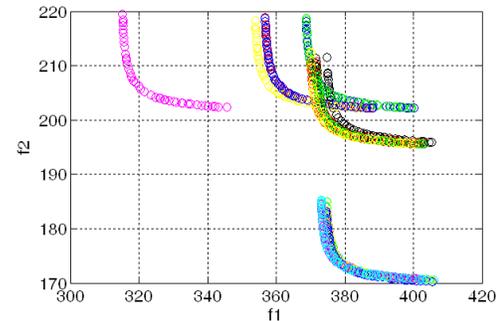


Figure 7: The evolved PL

b. The vessel-boats example

Here the vessel-boats (see Section I) example is solved in order to find the optimal strategies for the vessel. The genetic code encodes the different strategies outlined in Section I. First let us examine one possible strategy for the vessel, in which the vessel starts just left of the target, begins moving immediately when the game commences and intercepts the boat closest to it. Figure 8 depicts the results of three different games, each involving a different strategy for the boats. The strategies adopted by the boats differ in the number of "sacrificed" boats, namely one, two and three in the left, middle and right panels of the figure, respectively.

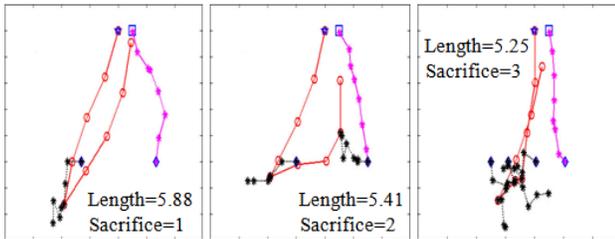


Figure 8: Three different games for the same strategy adopted by the vessel, showing that shorter distances are gained by increasing the number of sacrificed boats.

When only one boat is sacrificed (starting at the location marked by a diamond symbol in left panel), the sacrificed boat maneuvers so as to attract the vessel to it and away from the other boat, which heads towards the target (path designated in purple). The red path shows the path of the vessel. The evolution leading these results is described in [15]. As more boats are sacrificed, the length of the purple path gets shorter (the length of this path is indicated in each panel). The initial population for one strategy of opponent 2 is depicted in Figure 9.

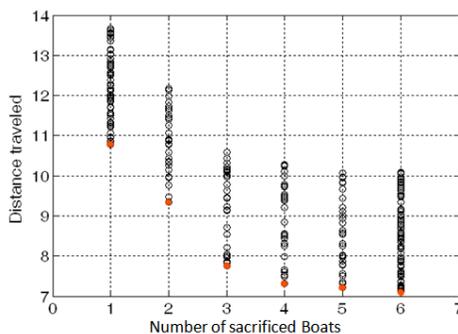


Figure 9: Initial population of strategies that may counter the strategy of one vessel (black circles) and the related evolved front (bold red circles).

The result of searching for this strategy's related front (Equation 4) is also depicted in the figure, designated by bold red circles. Figure 10 depicts three Pareto fronts (designated in different colors) that belong to the PL, each representing an optimal strategy for the vessel. These fronts are obtained by running the evolutionary algorithm (Section III.E.4).

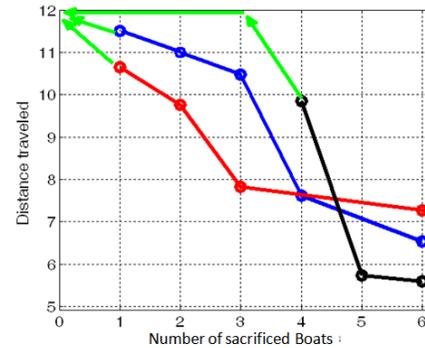


Figure 10: Three fronts of the PL, each associated with a different strategy for the vessel. Choosing each of the related strategies has pros and cons with respect to the problem's objectives.

The green arrows indicate a length of infinity. This means that none of the boats managed to reach the target. Therefore, if the vessel chooses the black-related strategy, at least three boats must be sacrificed in order for one of them to reach the target. This strategy is the one in which the vessel starts close to the target and intercepts the boats nearest to it, starting when the distance between it and the nearby boat is less than three units. However, if the vessel adopts the same strategy and four boats are sacrificed, one boat may reach the target point in a relatively short time (distance). This highlights the no-free-lunch decisions associated with players in MOGs.

V. SUMMARY AND CONCLUSIONS

The current paper makes two main contributions to the state of the art of MOGs. The first involves the definition of the PL, which is a rational representation of optimal strategies for MOGs. It is rooted in the idea of Pareto in the sense that for each strategy adopted by one opponent, there may be a set of optimal strategies to counter it. Until now, arbitrary equilibrium points may have been found while this important aspect was overlooked. Moreover, the currently solved problems are confined to rather simple continuous problems. This drawback highlights the second contribution of the current paper, an initial attempt to solve MOGs using artificial intelligence tools. Such an approach paves the way for solving difficult MOGs. Specifically, we suggested an MOEA which evolves the PL. For the evolution, a fitness measure has been suggested. It assigns high fitness to dominating convex sets. Each set represents optimal strategies adopted by one opponent to counteract one strategy adopted by the other opponent. The evolved PL is a set of sets. It is a representation of the performances of optimal strategies that may be adopted by one opponent (which maximize the MOG's objectives), each associated with a set of optimal strategies that may be adopted by the other opponent (which minimize the MOG's objectives). Two examples were used to demonstrate the algorithm. Both highlighted the advantage of utilizing the PL to support decisions while playing MOGs. Using the PL, an opponent

may constrain the multi-objective performances of the other opponent to a preferred region within the objective space by choosing one strategy from the Pareto set.

VI. FUTURE WORK

We believe the findings of this paper may shed new light on how MOGs are considered and solved. Therefore, it is clear that future work should include a wide range of topics, some of which are listed below.

1. As explained in Section III.D, the evolved PL may serve one of the opponents. This opponent is assumed to begin the game, thus constraining the other player. If support for the other player is to be considered, the optimization should be changed, leading to a different PL. A system for supporting players in making decisions based on the two PLs should be studied.
2. The proposed algorithm is not limited by the number of objectives, and MOGs with more than two objectives should be also investigated, for example, by adding radar to the vessel-boats problem with the objective of maximizing the boats' exposure to radar (or minimizing exposure when taking the boats' strategies into consideration).
3. The problem of embedding a MOEA that searches for the Pareto front for each strategy and then evolving these fronts towards the PL is associated with high computational complexity, as discussed in [13]. This complexity might be reduced by taking a co-evolutionary approach [18], which has been suggested for this purpose.
4. Although the importance of MOGs has been highlighted in several papers (e.g., [1]), more real-life examples should be introduced and solved.
5. MOGs should be used as benchmarks for comparing algorithms. At least one should be rather simple, so that the results can be verified with respect to exact solutions obtained through, e.g., mathematical programming.
6. The proposed approach is stochastic; therefore a statistical analysis of the results should be conducted, especially as related to repeatability.
7. The use of MOGs may make game-competitions even more exciting.
8. As indicated in Section II.C, learning strategies within MOGs have not been considered before. This is not surprising, considering that the current work seems to be the first to use artificial intelligence for solving MOGs.

ACKNOWLEDGMENTS

This study has been supported by Ort Braude College of Engineering grant number 5000.838.1.4-11.

REFERENCES

- [1] K. Somasundaram, and J. S. Baras, "Pareto Nash Replies for Multi-Objective Games". The Institute for Systems Research ISR technical report 2009-4, 2009.
- [2] J. Zhao, "The equilibria of a multiple objective game. International Journal of Game Theory, 20:171-182, 1991.
- [3] A. Zelikovsky, L. Dmitrii, D. Solomon, "Multiobjective games and determining Pareto-Nash equilibria". BULETINUL ACADEMIEI DE

- STIINTE, A REPUBLICII MOLDOVA MATEMATICA, (3):115-122, 2005.
- [4] M. Zeleny, "Games with multiple payoffs". International Journal of Game Theory, 4(4):179{191, December 1975.
- [5] B.M. Contini, "A decision model under uncertainty with multiple payoffs". In A. Mensch, ed., Theory of Games; Techniques and Applications, pages 50-63. American Elsevier Pub. Co., 1966.
- [6] Sakawa, M., (2002), *Genetic Algorithms and Fuzzy Multiobjective Optimization*. Kluwer Academic Publishers, Boston–Dordrecht–London.
- [7] D. Van Veldhuizen, and G.B. Lamont, "Multiobjective evolutionary algorithms: Analyzing the state-of-the-art". Evolutionary Computation, 8(2), pp. 125-147, 2000.
- [8] K. Deb, "Multi-objective optimization using evolutionary algorithms." J. Wiley & Sons, Ltd., 2001.
- [9] E. Zitzler, and S. Künzli, "2004. Indicator-Based Selection in Multiobjective Search". In X. Yao et al., editors, Conference on Parallel Problem Solving from Nature (PPSN VIII), volume 3242 of LNCS, pp: 832–842, Springer, Berlin.
- [10] E. Zitzler, E., L. Thiele, M. Laumanns, C. M., Fonseca, and V. G. Fonseca, "Performance assessment of multiobjective optimizers: An analysis and review". IEEE Transactions on Evolutionary Computation 7(2), 117–132. 2003.
- [11] J. Bader, D. Brockhoff, S. Welten, and E. Zitzler "On Using Populations of Sets in Multiobjective Optimization," In Ehr Gott M., et al., editors, 2009. Conference on Evolutionary Multi-Criterion Optimization (EMO 2009), volume 5467 of LNCS, pp: 140–154. Springer, Berlin, 2009.
- [12] G. Avigad, E. Eisenstadt and A. Goldvard, "Pareto Layer: Its Formulation and its Search by way of Evolutionary Multi-Objective Optimization." *Engineering Optimization*, 42(5): 453-470, 2010.
- [13] G. Avigad and J. Branke, "Embedded Evolutionary Multi-Objective Optimization for Worst Case Robustness". In *2008 Genetic and Evolutionary Computation Conference (GECCO'2008)*: 617-624, 2008. ACM Press, July 2008, Atlanta, USA (ISBN 978-1-60558-131-6).
- [14] Agapitos, A et al., Generating Diverse Opponents with Multiobjective Evolution. In Proceedings of the 2008 IEEE Symposium on Computational Intelligence and Games, pp: 135-142, Perth, Australia, 2008.
- [15] G. Avigad, E. Eisenstadt and M. Weiss, "The Optimization versus Survival Problem and its solution by an Evolutionary Multi Objective Algorithm". In the proceedings of SEAL 2010, LNCS 6457, pp: 494-503, 2010.
- [16] J. Neumann and O. Morgenstern, "Theory of games and economic behavior". Princeton, Princeton Univ. Press. 1953.
- [17] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.
- [18] J. Branke and J. Rosenbusch, "New Approaches to Coevolutionary Worst-Case Optimization". Parallel Problem Solving from Nature – PPSN X, Lecture Notes in Computer Science, 2008, Volume 5199/2008, 144-153, DOI: 10.1007/978-3-540-87700-4_15