The Sequential Optimization-Constraint Multi-objective Problem and its Applications for robust Planning of robot paths

Gideon Avigad and Kalyanmoy Deb

***Abstract*— In this paper a new approach to search for diverse solutions for a multi-objective problem is presented. Commonly, a search for solutions for a multi-objective problem, which is aimed at optimization, results in a set of Pareto optimal solutions. There are cases where more solutions should be also considered, nonetheless preserving the optimization inspiration. These solutions should not resemble the Pareto set, so as to provide diversity within the design space, and therefore they might not always be found by taking an epsilon-Pareto approach. With this motivation in mind, an already established method, which searches for diverse solutions, which are not all necessarily optimal, is herewith discussed and its shortages are highlighted. In contrast to the already established design method, the approach taken in this paper is to solve the multi-objective problem repeatedly, adding (automatically or interactively) at each run constraints, which are constructed, based on the obtained Pareto set. The motivation for the introduced approach comes from the need to generate a set of robot paths, which allow a mobile robot operator, flexibility in complying with different planning demands and a rapid response to a developing scenario. The methodology and the applicability of the approach are explained and demonstrated by utilizing multi-objective path planning problems.**

I. INTRODUCTION

SEARCHING for solutions to Multi Objective Problems (MOPs), commonly involves a search for the optimal solutions for the problem. Whenever the objectives of such a problem are contradicting, the solutions are termed Pareto optimal solutions, and their performances belong to the Pareto front [1]. The selecting of a solution out of such a set of solutions is based on the designers' preferences.

Producing a variety of optimal solutions within the framework of Evolutionary Computation (EC) is done by striving to diversity. This is done by applying a pressure towards such a diversity during the evolution. The diversity might be aimed at the decision space (e.g., [2]), the objective space (e.g., [3]), or at both spaces simultaneously (e.g., [4]).

For supporting designers' decisions, it is important not to flood the designers with an exaggerated number of Pareto-optimal solutions (e.g., [5]). To dilute the number of solutions, several clustering approaches have been suggested. Such a clustering is commonly performed based on the objective space (e.g., [6]). These approaches allow the presentation of distinct solutions, each possessing a distinctiveness, which deviates them one from the other. Such deviations make the human decision process easier.

In contrast to the above (the problem of overflow of Pareto-optimal solutions), this paper deals with cases where there is a lack of distinct Pareto solutions to present to the designers/planners. Such a lack has been treated in [2]. The lack may be avoided by searching for distinct less optimal solutions. To elucidate the sense in conducting such a search, refer to Figure 1. In the figure, a MOP's bi-objective space is depicted. Also depicted are the performances of ***all possible*** solutions of the MOP.

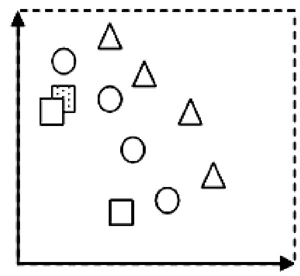

Figure 1: Entire set of solutions' performances for a possible contradicting bi-objective problem.

Considering a min-min problem, an optimization procedure would result with just two distinct solutions, which are designated as blank squares in the figure. Now suppose that more distinct solutions are needed. One approach that might be considered is to relax the demand of optimality by taking an epsilon-dominance approach (e.g., [7]). Such an approach does not guarantee finding new distinct solutions in the design space and similar to the Pareto-optimal solutions may be found (e.g., the doted square).

So, the approach is to look for new distinct solutions, which are less optimal (i.e., *dominated* solutions). Good candidates for such a search, assuming their diversity in the design space, could be the solutions shown in circles and then the triangles, which are shown in the figure.

In this paper a formulation of the MOP, which suits such a search, is provided. Moreover a formulation for a proposed solution to the problem is also given. In contrast to the approach of [2] a sequential approach to search for the solutions to the problem is given. Such a different approach has advantages over the former as related to computational time and appropriateness to path-planning problems.

This paper is organized as follows. Section II, gives the background to the relevant issues of this paper including MOPs, EMO, diversity-preservation, distinctiveness, EMOs for dynamic environments and multi-objective path planning. Section III, includes a discussion on the drawbacks of existing approaches and gives the motivation for this paper approach. Section IV, introduces the methodology, which includes the relevant formulations, and the EMO approach. Section V, demonstrates the applicability of the methodology by utilizing path planning problems.

G. Avigad is with the Aby and Aldar faculty of Engineering at Tel Aviv University, Israel. (e-mail: gideon@eng.tau.ac.il)

K. Deb holds Shri Deva Raj Chair Professor at the Indian Institute of Technology Kanpur, India. (e-mail: deb@iitk.ac.in)

1-4244-1340-0/07$25.00 ©2007 IEEE

Conclusions and future-work are given in section VI.

II. Background

Multi-objective search is an important research topic. It concerns the search for solutions to real world problems, which many of them are MOPs. In case of contradicting objectives there is no universally accepted definition of an 'optimum' as in a single-objective optimization ([8]). In such a case, there is no single global solution and it is often useful to determine a set of solutions that fits a predetermined definition for an optimum and let the decision-maker choose between them. The predominant concept in defining such a set is that of Pareto optimality ([1]). By definition, Pareto-optimal solutions, which belong to the Pareto optimality set, are considered optimal because there are no other designs that are superior in all objectives (e.g., [9]). The search for optimal solutions for a MOP is commonly termed Multi Objective Optimization (MOO).

A comprehensive survey and comparison between most multi-objective search techniques and algorithms can be found in [10].

Evolutionary algorithms possess several characteristics, which make them suitable for solving MOOPs [11]. According to a recent review by Coello, [12], Evolutionary Multi-objective Optimization, (EMO), has reached a matured stage, and its development has consistently been followed by applications in engineering, product development, management, and science. The development of Pareto-based evolutionary algorithms has been initiated by the procedure suggested by Goldberg, [13]. Surveys and descriptions of such algorithms can be found in several references (e.g., [14]).

In EMO the algorithms should find all the trade-offs among the conflicting objectives. Therefore, ensuring diversity along the front is a must for any successful Multi-Objective Evolutionary Algorithm (MOEA). A recent review, which has been conducted in [11], classifies existing methods for diversity preservation according to the three categories of statistical density estimation including: kernel methods, nearest neighbor, and histogram techniques. Fitness sharing, which is a popular technique for diversity preservation in MOEA, falls into the first category (e.g., [15]). MOEAs commonly use sharing as a mean to equally distribute the vectors, which approximate the Pareto front. Preserving diversity within the design space has also been treated (e.g., [16]. Different variants of preserving diversity have been employed, as reviewed in [8], with both genotypic and phenotypic distance measures.

The above classified approaches are associated with the implanting of diversity preservation mechanisms within the evolutionary search. Such implanting may be viewed as a-priory diversity preservation approaches. This is due to the fact that the rules of what is different are embedded within the search algorithm.

Posteriori diversity approaches may also be found. Such approaches may be classified under the clustering paradigm.

In multi-criteria optimization, data clustering can be a useful exploratory technique in knowledge discovery. Since it groups similar solutions together, it allows the decision-maker to identify potentially meaningful trade-offs among the solutions contained in the Pareto-optimal set without requiring the decision-maker to explicitly define objective function weights or utility functions ([5]).

Cluster analysis is a multivariate analysis technique that is defined as the process of organizing objects in a database into clusters/groups such that objects within the same cluster have a high degree of similarity, while objects belonging to different clusters have a high degree of dissimilarity (e.g., [5]). The most popular nonhierarchical clustering method is probably the *k*-means clustering algorithm. The *k*-means algorithm is well known for its efficiency in clustering data sets. The grouping is done by calculating the centroid for each group, and assigning each observation to the group with the closest centroid. More elaborated approaches, which utilize EMO also exist (e.g., [17]).

Another posteriori approach within the framework of EMO has been suggested by Deb (e.g., [6]). In this approach a learning algorithm is applied to identify distinct designs. Such learning may give an insight as for the relation between the design space and the objective space, thus supporting decision making.

Often, algorithms provide solutions that may not be Pareto optimal, but may satisfy other criteria, making them significant for practical applications. In such a case solving a MOP problem is not equivalent to solving the MOOP, which may also be defined for that problem. For example Parmee ([18]), introduced Cluster Oriented Genetic Algorithm (COGA), where the result of the MOP search is a set of solutions that are related to 'interesting regions'. Another approach is the goal attainment approach ([19]) that focuses on finding a solution (s) around a target (goal) in the objective space.

With respect to this paper focus, a most important study has been conducted in [2]. In [2] the Clustering Pareto Evolutionary Algorithm (CPEA) has been suggested. In the same manner that niching preserves diversity of multi-modal functions' solutions within a single objective problem, the CPEA searches for diverse solutions within MOPs including solutions that are not Pareto optimal. The authors of [2], aimed at finding several *local* Pareto-optimal sets. They referred *local* to decision variable space – technology type and operating conditions. According to [2], the advantage of multimodal optimization is that the engineer is able to choose from a set of local optima a solution that is perhaps not the *global* optimum of the system defined by the system model and objectives, but which may be the *best* solution of the real system when other, difficult to quantify considerations are taken into account. Such an approach implies on multiple Pareto optimal sets in objective space, that may overlap, or where one local set may be entirely dominated by another, which represents different niches in the decision space. According to the CPEA procedure, an initial population is randomly generated and the objectives for each individual are evaluated. Thereafter the solutions are clustered, and the individuals in each cluster are given a rank based on how many other individuals in the cluster dominate or are dominated by the point. In the next step each cluster is given the chance to produce a predefined number of children. This is done through a real variable crossover mechanism. The following steps are associated with the insertion of the children to the population and the "cleaning" of the population from less successful results from each cluster. The result of the CPEA are clustered sets of solutions, each representing a distinct design, having its performances on the Pareto front or on a local front.

Apart from the above approach another set of approaches, which are associated with dynamic multi-objective problems, is of interest to this paper study. Dynamic multi-objective problems are multi-objective problems with time depended objective landscape, time depended solution landscape or a combination of them (see [20] for a detailed classification). According to [21], the existing EC approaches to solve a dynamic MOP may be classified into two main categories. The first category deals with the control of the two basic functions of the algorithm's population in a dynamic environment: converging on the current global optimum and exploring the design space for the optimum's next location or for new optima as soon as the objective landscape changes. These two functions usually compete against each other and this competition can be viewed as a balance between convergence and diversity. An example to an approach within this category is the Self-Organizing Scouts by Branke et al [22], which separate the population into groups with specific functions of either tracking an optimum or exploring for new solutions. The second broad category of approaches is concerned with exploiting past information (past fit solutions) which might again become useful as the problem evolves. An example for such exploitation of past information has been demonstrated by Branke et al. [23] in their discussion of changing environments.

When considering the use of the approaches used to solve dynamic MOPs, it is important to note that the nature of the time-changing environment is an important aspect (e.g., [21]). Two basic categorizations of dynamic environments are the *frequency* and the *severity* of change (e.g., [23])).

Although path planning problems should be investigated within the framework of a multi-objective problem, searching for Pareto solutions for path planning problems is a new research area. Such investigations include [24], [25] and [16]). In [16], the ideas presented in [2] were implemented leading to finding both optimal and non-optimal paths. Non-optimal paths were also considered in [21], where human preferences towards conceptual paths influenced the search. Nonetheless the approach taken in this paper is totally different and addresses other aspects of the planning. Moreover it possesses a different motivation for the search as explained in the following.

III. MOTIVATION

Observing the reviewed studies as surveyed in the background, it may be acknowledged that in the CPEA there is an inherent a-priory conviction that there is a lack of solutions and therefore the entire space should be searched. This means that all possible designs (as the number of predefined clusters) are all to be found. As noted in [2] such a-priory determination of the number of clusters is related to a computationally expensive procedure. Moreover in the CPEA, no priority is given during the search to more optimal clusters. This means that if the number of allowed clusters is high enough, it might happen that also "very bad" solutions would evolve.

The above observations call for a more persistent search. Initially, it should be checked if there are enough Pareto solutions. If the answer is that there are not enough such solutions, only then further solutions should be searched for, maintaining the optimality and diversity motivations. Moreover, observing the evolved Pareto front and its related Pareto set, may give more insight with relation to the missing solutions.

To further elucidate the above, refer to Figures 2a and 2b. Each of the figures depicts the performances of all possible distinct solutions within a bi-objective space belonging to different problems. Solutions from different levels of non-dominance are designated by different symbols in both figures. Suppose that 10 different solutions (associated with design space clusters) are searched for by CPEA, in each of the figures. The resulting performances of the found solutions are designated by blackened symbols. In the first case (Figure 2a) it is evident that the Pareto solutions span the objective space and there is no obvious need for more solutions than those, which are designated by squares. If such a need arises the circles and triangle related solutions are good candidates. *Nonetheless the diamond related solutions would have also been found by CPEA, although they possess profoundly worse performances than the Pareto front.*

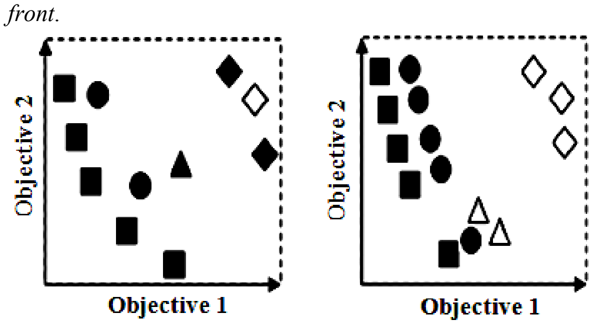

Figure 2a and 2b: Two cases of performances in a bi-objective space. Squares designate the Pareto front while circles, triangles and diamonds designate lower levels of non-dominance.

In the second case (Figure 2b) *it seems that finding the four circle related solutions, which span solutions with good performances in objective 1, is somewhat a waste of computational resources.* It would have been wiser to search for solutions with good performances with respect to objective 2, as they are scarce.

When considering dynamic EC approaches to attend the above problem, it is noted that according to [21]: "*if the severity of change is especially large, one instance of the problem will be completely unrelated to the next and it can be argued that a complete re-start of the algorithm is as efficient as any dynamic optimization methodology*." It seems that the dynamic changes associated with the problems attended by the current paper, involve such severe cases. A shift from one map setting to another is a discrete jump, which seems to be hard to follow by a dynamic approach. Moreover the purpose of dynamic MOP solving approaches is to track the changing environment, whereas here, the motivation is finding diverse solutions. It is further noted that the human interactivity within the search process of dynamic MOPs solving approaches does not exist. Here such interactivity is of a major interest.

From this discussion it may be realized that a stepwise approach should be adopted. That is, to develop one front at a time, while inserting constraints to direct the search to find more solutions associated with observed lacks. Such an approach is developed in the next section.

Observing the existing studies, which are related to path planning within the framework of EMOs, it is realized that robustness, from any kind, has been scarcely treated. This means that neither robustness to changes in the objective functions, nor robustness to uncertainties associated with environmental parameters, has been fully explored. Such robustness consideration may affect the location of the frontier. In other words solutions that are not Pareto-optimal may be considered for selection as they are more adequate than the Pareto-optimal set (see e.g., [26]), e.g., when robustness is thought. Such a situation is most relevant to the current motivation, which is focused on searching for solutions that are not just from the Pareto-optimal set.

In this paper one kind of robustness is considered and used as a motivation to search for other solutions than the Pareto-optimal paths.

IV. METHODOLOGY

A. The robust planning MOP

The robustness in the current investigation is associated with the robustness of the planning to changes in the problem conditions. Such changes are dependent on a possible developing scenario or on designers imposed conditions. To elucidate the problem, refer to Figure 3. The figure depicts a path planning map with a start and target points as well as some obstacles. The MOP of the related planning task is to minimize both the travel distance of a mobile robot that is to move from the start to the target, as well as to minimize its maneuvers (the sum of angle changes). It is clear that the strait path between the start and the target is the optimal path as it is the shortest and is associated with a travel with no maneuvers. This path is the one, which would have been found by an optimization procedure. Now suppose that within the map there is a dynamic opponent, which may exist down to the doted horizontal line, and having a width of the gap between the upper left most and the upper middle obstacles. If the opponent blocks the gap, alternative travels should be searched for, barring in mind the optimization objectives. Good candidates are the paths which are designated by dots (left side and right side) and the path that is designated by a line-dot designation.

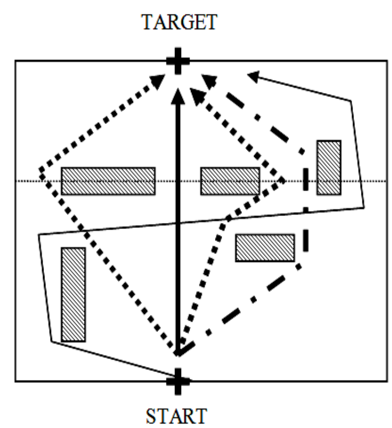

Figure 3: An example demonstrating the motivation for the current approach

Each has its advantages. The right-located doted path is the shortest (not considering the strait optimal path, which is now constraint) nevertheless it is associated with two profound maneuvers. The left-located doted path is longer than the former but has fewer maneuvers. The line-dot path is longer than the other two but it is associated with the least harsh maneuvers. This means that there are three Pareto-optimal solutions in a problem that excludes the straight path. Now suppose that it is possible that there may be more than one opponent. This may lead to a need to consider the zigzagged path, which is longer and is associated with more intensive maneuvers than all former found paths.

It is noted that path planning may be involved with many different robustness related cases which may call for the need to find not just the optimal set. Some different cases, which may be treated by the approach introduced in this paper, are given in the following section.

A-priory planning of different path plans, which takes into account the robustness as explained above, will permit a rapid decision making and execution of different optimal paths by moving between ***pre-planned optimality-based paths***.

B. Possible robustness related constraints

It is suggested here, that the search of solutions, which are not part of the Pareto-optimal front may be directed by three different constraints. Each of these constraints is associated with a different robustness related problem. These different cases are listed and explained in the following.

1. *A distinct path (solution) is searched for*. The extent of the difference between such a solution and previously found ones (in the design space) may be controlled. A search for paths x, $x \in X \subseteq S \subseteq R^n$ is subject to: $(x - x^{(s)}) < \delta$, where X is the feasible decision space, $x^{(s)}$ is a previously found solutions and δ is the permitted difference between the searched paths and previous found ones (in the design space).

An example for this case is when it is desired to send equipment from the start point to the target, each time altering the path, thus avoiding detection by avoiding repetition. Such a case is elaborated in section V.

2. *Uncertainty as related to the location of dynamic obstacles*. In such a case the constraint, $h'(x) = 0$ limits the search space to exclude "dangerous" regions. It is noted that such constraints are added to the problem definition posteriori to finding the Pareto set. The prime assigned to h denotes its being an added constraint. An example for this case has been given in subsection IV.A, with relation to figure 3.

3. *Lack of solutions within a region of the objective space*. Constraining the objective space by $F'(x) < 0$ limits the search of solutions to those with performances that belong to unconstraint sub-spaces of the feasible objective space. The prime assigned to F denotes its being an added constraint. Such constraints are added to the problem definition posteriori to finding the Pareto-optimal set. This may be done following human inspection of the obtained Pareto-optimal front or automatically by a computer search.

An example for such a case may be understood with a context to the example in section III with relation to Figure 2b.

It is noted again that the mandatory motivation is optimality and therefore Pareto related paths are searched initially. The non-optimal paths are therefore found later but preserving the optimality inspiration.

C. Problem definition

The problem of finding distinct solutions is defined by a sequence of problems, each with new constraints imposed based on the solution for the former solved problem (but the first one).

1. Find $x^{(1)}$ which solves the problem: $\min F(x)$ s. t. $x \in X \subseteq S \subseteq R^n$ and h(x)=0.
2. For all i=2, …., n find $x^{(i)}$ which solves the problem: $\min F(x)$ s. t. $x \in X \subseteq S \subseteq R^n$, g(x)<0 ,h(x)=0 and/or $\bigcup_{s=1}^{i-1}(x - x^{(s)}) > \delta$ and/ or $h'(x\ ) = 0$ and/or $(F'(x\ )) < 0$.

Where n is the number of times the problem has to be solved before sufficient solutions are obtained and δ is the permitted difference between design variables. F(x) is $F(x) = [F(x_1),....F(x_K)]$ and K is the number of the problem objectives.

The above problem definition suggests that the problem of finding diversified solutions based on optimization is decomposed into a sequence of optimization problems. The initial optimization problem (step 1 above) is associated with the minimization of objective functions ($\min F(x)$) subjected to constraints of the paths. These initial constraints limit the paths search space ($x \in X \subseteq S \subseteq R^n$) to feasible paths and constrain the paths not to go through obstacles (h(x)=0). The following problems (step 2 above) are defined such the initial problem is solved but with added constraints. These constraints include the constraints which are described in section IV.B.

D. Problem Solution

The solution to the problem is n sets of Pareto-optimal solutions with a corresponding Pareto-optimal front for each. Each of the Pareto-optimal sets is optimal with respect to the solved problem and therefore all are optimal solution within the context of the problem solved. Naturally, in the overall problem there is just one Pareto-optimal set and front (the result of the first time the problem is solved). To prevent confusion between the different Pareto-optimal sets a different terminology will be used for each. The terms i-Pareto set and i-Pareto-optimal front correspond to Pareto-optimal sets and fronts, where i denotes the problem number, i=1,… n.

The solution to the problem is therefore optimal sets O^* and sets of Pareto-optimal fronts OF^*

$$O^* = \bigcup_{i=1}^{n} x^{(i)}$$

$$\text{where } x^{(i)} \therefore | \neg\exists\, x' \in X\ : F(x') \preceq F(x^{(i)}) \}$$

$$OF^* = \bigcup_{i=1}^{n} y^{(i)}$$

$$\text{where } \{ y^{(i)} \in Y \,|\, y^{(i)} = F(x^{(i)}) : x^{(i)} \in O^* \}$$

The formulations suggest in fact that the solution to the problem is sets of Pareto-optimal solutions, where each set solves a problem, which has its own constraints.

E. Solution Approach

The procedure of searching the solutions is outlined in the following:

1. Run an EMO algorithm to find the Pareto-optimal set for the problem.
2. Cluster the paths according to any preferred distance in the design space to find a set of distinct optimal paths.
3. Constrain the optimal set/front by any of the

constraints described in section IV.B.

4. While insufficient solutions are found go to 1.

The approach to repeat the evolution until enough solutions are sort, as realized by this algorithm, highlights again the view of the current paper that: *The global optimal set has to be found first, followed by sequential searches for local Pareto-optimal sets, based on robustness related needs.*

V. EXAMPLE

The following example demonstrates the methodology, which is given in section IV.

The problem, which serves for the example, is depicted in Figure 4 and is associated with a travel of a mobile robot from a start point (see point (5, 0)) to the target (see point (5, 10)) within a 10 by 10 grid map. The objectives of the path planning problem are to minimize the travel distance while minimizing visibility to an opponent located at the target. The area covered by the site of the opponent is gray colored. It is assumed by the planners, that after one use of a path it may be detected by the opponent and therefore the path should not be repeated more than delta percent. This means the planning should provide robustness in the sense that there are several optimality-based "safe" alternative paths to use.

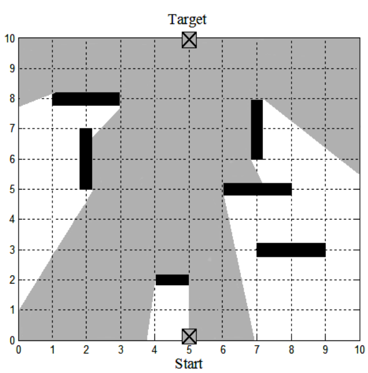

Figure 4: The multi-objective path planning problem

The steps of the algorithm, which has been introduced in section IV.D were followed to solve the problem. The paths have been coded by using three 8 bit binary strings for each path, each designating a point along the path. In the current program (used also for other applications) the 'y' points' locations are limited to be on the grid. The population contained 20 individuals which were run using the NSGA-II EMO algorithm, [27], for 150 generations using a two point crossover (50% rate) and mutation (3% rate). At the end of the first run (see step 1 in section IV.D), the paths were clustered to gain the Pareto-optimal set, which is depicted in Figure 5a. In the current problem the clustering needed, has been minor as the elite population converged to several distinct path solutions. Clustering in the current problem has been practiced by integrating the area between the paths. Paths close by, should have low values of the integral no matter where the decoded path points are. Two paths with an integral values of less than 1 in between, where considered as the same path with the best (according to non-dominance level and crowding) representing them. The 1-Pareto-optimal front of the problem is designated by circles in Figure 6.

Following the motivation not to repeat paths to a certain extent, a repetition boundary has been designated around each Pareto-optimal set path. These boundaries are designated by gray thick lines around the Pareto-optimal paths, in Figure 5b. In the next run, paths that existed within these boundaries for more than $\delta = 7\%$ were penalized (assigning them with the lowest non-dominance level of the generation).

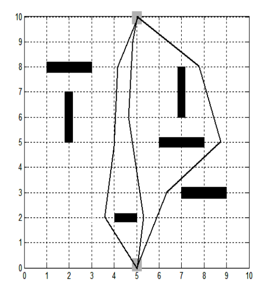

Figure 5a: 1-Pareto-optimal set

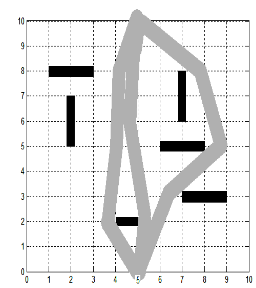

Figure 5b: constrained paths

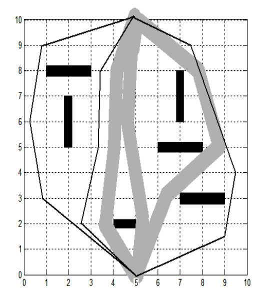

Figure 5c: 2-Pareto-optimal set

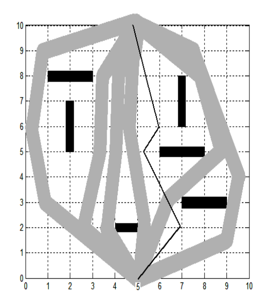

Figure 5d: 3-Pareto-solution

The resulting paths and the resulting 2-Pareto-optimal front are depicted in Figure 5c and as squares in Figure 6 respectively. Repeating the procedure once more, (the penalized area is shown by gray in Figure 5d, results in one optimal path, which is depicted in that same figure. The performances of the solution belong to the 3-Pareto-optimal front, which is depicted as a triangle in Figure 6. The three sequential runs produced 7 different paths, organized in three Pareto-optimal fronts. It is noted that these solutions diverse one from the other in both the path planning space (design space) as well as in the objective space. It is noted that the aimed diversity is within the design space and not in the objective space, although such dual diversity is welcomed. Each front is a result of an optimization procedure for a different setting associated with the same problem. The 7 paths may now be considered by the planners. When Pareto-optimal optimality is considered, there are three optimal paths and any of them could be used. Executing one of the 1-Pareto-optimal related paths does not mean that the next selected path should be chosen from that 1-Pareto-optimal set. For example if the lowest circle related solution is chosen to be executed first, then the next execution might be the path which is associated with the lowest square (belonging to the 2-Pareto). Such a decision may be taken as a result of the planners preferring low visibility paths over global Pareto-optimal paths.

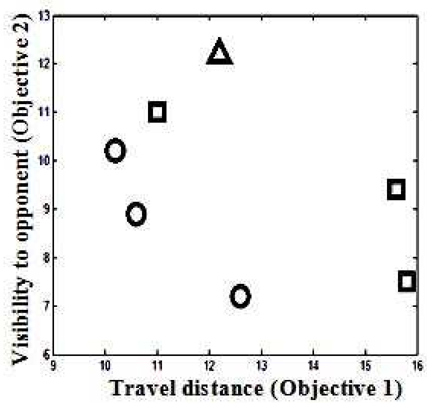

Figure 6: The three Pareto-optimal fronts obtained by three sequential runs. Circles designate the 1-Pareto-optimal, squares the 2-Pareto-optimal and a triangle for the 3-Pareto-optimal solution.

In any case the planners are exposed to a set of diverse paths, allowing them flexibility in overcoming unexpected hazards (obstacles) and a fast response to a developing scenario.

VI. DISCUSSION AND FUTURE WORK

Dealing with robustness in a multi-objective path planning problem setting is scarcely treated by EMO approaches. This paper introduced several MOPs, which highlight the need for robust path planning. Here the robustness is not associated with the robustness of one path solution or the other, but rather with the robustness of the planning process. Robustness of the planning means that the search for paths results in optimality-based diverse solutions, which allow flexibility in responding to changes in the path planning map settings. To allow such robustness it is important to search the path-plan space for solutions, which are diverse in both the objective space as well as within the design space. Such diversity may be at the expense of optimality but by that, should provide robustness in solutions. The diversity within the objective space allows the planners to choose different performance related paths as a reaction to changes and developing scenarios. The diversity within the design space allows rapid action to overcome dynamic changes of the map.

In this paper a straightforward approach to evolve such diversity-based path plans has been suggested. The suggestion included the formulation of the problem in hand and the formulation of the solution approach, which are realized through the implementation of an EMO algorithm.

In the proposed solution approach, the diverse path plans are obtained by solving a sequence of sub-problems, each constrained by its former. Considering the constraints, the paper suggests three possible kinds of robustness that might be treated by the suggested approach.

The result of running the algorithm according to the suggested approach enables us to find diverse sets of paths, where each set is associated with a different Pareto-optimal front and a different Pareto-optimal set.

It is the authors' view, that all the paths should be embedded with a robot-navigation software, allowing a shift from one path to another based on the robot operator's real time decisions.

Observing the hereby presented paper, it may be depicted that "path plans," "solutions" and "designs" are alternatively used for the same purpose. These multiple notions for the same concept may imply on the generic nature of the approach introduced in this paper. Therefore, as future work, the approach should be tested with relation to engineering design, although the motivation for considering non-optimal solutions is expected to be different. Moreover the approach may be used to test the aptitude of suggested EMOs to evolve diverse solutions in a single run. This means that this paper's approach may be utilized to find the diverse solutions sequentially, setting the test cases for simultaneous approaches.

Another investigation that should be considered as future work, concerns a comparison between this paper's approach and approaches suggested within the framework of dynamic multi-objective optimization and with the CPEA algorithm.

VII. REFERENCES

[1] Pareto, V. 1906: Manuale di Economica Politica, Societa Editrice Libraria. Milan; translated into English by A.S. Schwier as Manual of Political Economy, edited by A.S. Schwier and A.N. Page, 1971. New York: A.M. Kelley.

[2] Molyneaux, A. K.., Leyland, G. B., and Favret, D. A New, Clustering Evolutionary Multiobjective Optimization In *Proceedings of the Third International Symposium on Adaptive Systems-- Evolutionary Computation and Probabilistic Graphical Models*, Institute of Cybernetics, Mathematics and Physics, pages 41--47, Havana, Cuba, March 19-23 2001

[3] Zitzler, E. Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications. *Ph.D. thesis* submitted to the Swiss Federal Institute of Technology (ETH) Zurich, Switzerland, 1999.

[4] Chan, K.P and Ray, T. An Evolutionary algorithm to maintain diversity in the parametric and the objective space, *Proceedings of the third International Conference on Computational Intelligence, Robotics and Autonomous Systems,* (*CIRAS 2005*), Singapore, 2005.

[5] Taboada H.A., and Coit, D. W. Post-Pareto Optimality Analysis to Efficiently Identify Promising Solutions for Multi-Objective Problems, www.engr.rutgers.edu/ie/research/working_paper/paper%2005-015.pdf, 2005.

[6] Deb, K. Unveiling Innovative Design Principles by Means of Multiple Conflicting Objectives, *Engineering Optimization,* Vol. 35, No. 5, , 445–470, 2003.

[7] Laumanns, M.., Thiele, L., Deb, K., and Zitzler, E. (2001): On the Convergence and Diversity-Preservation Properties of Multi-Objective Evolutionary Algorithms. TIK-Report No. 108, Swiss Federal Institute of Technology (ETH) Zurich, Switzerland.

[8] Van Veldhuizen, D., and Lamont, G.B. Multiobjective evolutionary algorithms: Analyzing the state-of-the-art. *Evolutionary Computation*, 8(2), pp. 125-147, 2000.

[9] Meittinen, K. M. *Non-linear Multiobjective optimization*, International Series on Operations Research and Management Science. Kluwer Academic publishers, 1999.

[10] Marler, R.T., and Arora, J.S. Survey of multi-objective optimization methods for engineering, *Structural Multidisciplinary Optimization*, 26: 369–395, 2004.

[11] Zitzler, E., Laumanns, M., and Bleuler, S. A Tutorial on Evolutionary Multiobjective Optimization, in Xavier Gandibleux, Marc Sevaux, Kenneth Srensen and Vincent T'kindt (editors), *Metaheuristics for Multiobjective Optimisation*, pp. 3--37, Springer. Lecture Notes in Economics and Mathematical Systems Vol. 535, Berlin, 2004.

[12] Coello, C.A.C. Recent Trends in Evolutionary Multiobjective Optimization, in Ajith Abraham, Lakhmi Jain and Robert Goldberg (editors), *Evolutionary Multiobjective Optimization: Theoretical Advances And Applications*, pp. 7--32, Springer-Verlag, London, 2005, ISBN 1-85233-787-7.

[13] Goldberg, D. E. Genetic Algorithms in Search, Optimization, and Machine Learning, Addison Wesley, Reading, Massachusetts, 1989.

[14] K. Deb (2001). *Multi-Objective Optimization using Evolutionary Algorithms*. J. Wiley & Sons, Ltd.

[15] Van Veldhuizen, D., and Lamont, G.B. Multiobjective evolutionary algorithm test suites. *ACM*, 1998.

[16] Castillo, O. and Trujillo, L. Multiple Objective Optimizations Genetic Algorithms for Path Planning in Autonomous Mobile Robots, *International Journal of Computers, Systems and Signals, Vol. 6, No. 1, 2005*.

[17] Kaufman L. and Rousseeuw P.J. (1990). Finding groups in data. An introduction to Cluster Analysis. Wiley-Interscience

[18] Parmee I.C., The maintenance of search diversity for effective design space decomposition using cluster oriented genetic algorithms (COGAs) and multi-agent strategies (GAANT), *Proceedings of Adaptive Computing in Engineering Design and Control II*, University of Plymouth, pp: 128-138, 1996b.

[19] Gembicki, F.W. Vector optimization for control with performance and parameter sensitivity indices. *PhD. thesis, Submitted to Case Western Reserve University*, Cleveland, Ohio, 1974.

[20] Farina, M., Deb, K.., and Amato, P. Dynamic multiobjective optimization problems: Test cases, approximations, and applications. *IEEE Transactions on Evolutionary Computation*. 8(2004)5, 425–442.

[21] Hatzakis, I., and Wallace, D. Dynamic Multi-Objective Optimization with Evolutionary Algorithms: A Forward-Looking Approach, In Proceedings of the 8th annual conference on Genetic and Evolutionary Computation ({GECCO} '06), pp: 1201:1208, 2006.

[22] Branke, J., Kaussler, T., Schmidt, C. and Schmeck, H. 2000. A Multi-Population Approach to Dynamic Optimization Problems. In *Adaptive Computing in Design and Manufacturing 2000*, pp: 99-308, Springer.

[23] Branke, J., Salihoğlu, E., and Uyar, Ş. 2005. Towards an analysis of dynamic environments. In *Proceedings of the 2005 Conference on Genetic and Evolutionary Computation* (Washington DC, USA, June 25 - 29, 2005). H. Beyer, Ed. GECCO '05. ACM Press, New York, NY, 1433-1440.

[24] Mittal, S. and Deb, K. (in press). 'Three-dimensional path planning for UAVs using multi-objective evolutionary algorithms. Proceedings of the Congress on Evolutionary Computation (CEC-2007), 25-28 September, 2007.

[25] Moshaiov, A., and Avigad, G. Multi-objective Path Planning by the Concept-based IEC Method, *Proceedings of the 2004 IEEE International Conference on Computational Cybernetics, (ICCC 2004*), Vienna, Austria, 2004.

[26] Deb, K., and Gupta, H. Introducing robustness in multi-objective optimization. *KanGAL Report No. 2004016*, October, 2004.

[27] Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T., 'A Fast and Elitist Multiobjective Genetic Algorithm: NSGA–II,' *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, April 2002.