Visualizing high dimensional objective spaces for multi-objective optimization: a virtual reality approach

Valdés, Julio; Barton, Alan

National Research Council Canada
Conseil national de recherches Canada

Institute for Information Technology
Institut de technologie de l'information

NRC·CNRC

Visualizing High Dimensional Objective Spaces for Multi-Objective Optimization: A Virtual Reality Approach *

Valdés, J., and Barton, A.
2007

* Proceedings of the IEEE Congress on Evolutionary Computation. Singapore. September 25-28, 2007. NRC 49365.

Canada

Visualizing High Dimensional Objective Spaces for Multi-objective Optimization: A Virtual Reality Approach

J. J. Valdés and A. J. Barton

Abstract— **This paper presents an approach for constructing visual representations of high dimensional objective spaces using virtual reality. These spaces arise from the solution of multi-objective optimization problems with more than 3 objective functions which lead to high dimensional Pareto fronts which are difficult to use. This approach is preliminarily investigated using both theoretically derived high dimensional Pareto fronts for a test problem (DTLZ2) and practically obtained objective spaces for the 4 dimensional knapsack problem via multi-objective evolutionary algorithms like HLGA, NSGA, and VEGA. The expected characteristics of the high dimensional fronts in terms of relative sizes, sequencing, embedding and asymmetry were systematically observed in the constructed virtual reality spaces.**

I. Introduction

The role of visualization techniques in the knowledge discovery process is well known. The increasing complexity of the data analysis procedures makes it more difficult for the user to extract useful information out of the results generated by the various techniques. This makes graphical representation directly appealing.

In multi-objective optimization, rather than finding a single best solution for a given problem, what is found is a set of "compromise" solutions from which the decision maker selects a particular one, based on additional domain knowledge. For up to three objectives, a scatter plot suffices for displaying the set of solutions on which the decision will be made, but this approach is no longer possible when the problem involves more than 3 objectives. This prevents the decision maker from using visual information in the decision making process and therefore to use the powerful geometric pattern processing capabilities of the human brain.

This paper explores the construction of spaces for visual data mining of multi-objective optimization results (in particular, objective spaces) using virtual reality as the visualization environment. A first approach is presented showing properties of the multi-dimensional objective spaces. With virtual reality, the decision maker can not only visualize but also navigate and interact with the information. This approach is general in the sense that both spaces, the original and the virtual, might have been computed with or without evolutionary computation methods.

Two types of problems are illustrated: *i)* the representation of an increasingly high dimensional collection of theoretical Pareto fronts derived from a test problem (DTLZ2), and *ii)* the comparison of approximations to the Pareto front of a real problem, obtained with different evolutionary computation-based multi-objective optimization methods.

J. J. Valdés and A. J. Barton are with the National Research Council Canada's Institute for Information Technology's Integrated Reasoning Group, 1200 Montreal Road, Ottawa, Ontario, Canada, K1A 0R6 (email: julio.valdes@nrc-cnrc.gc.ca and alan.barton@nrc-cnrc.gc.ca).

II. Multi-objective Optimization

Multi-objective optimization (MOO) studies optimization problems involving more than one objective function and the goal is to find one or more optimal solutions. Most real world problems involve multiple objectives and typically different solutions lead to conflicting scenarios among the objectives under consideration: a solution which is optimal in the sense of a given objective might not be from the point of view of one or more of the other objectives [1]. Therefore, as the user typically has to chose only one solution, trade-offs are required and the goal is to find a set of optimal solutions representing the best trade-offs, from which the user, with further higher level information about the problem can make a decision.

Most multi-objective algorithms use the concept of dominance in their formulation. A solution $\overleftarrow{x}_{(1)}$ is said to dominate [2] a solution $\overleftarrow{x}_{(2)}$ for a set of m objective functions $< f_1(\overleftarrow{x}), f_2(\overleftarrow{x}), ..., f_m(\overleftarrow{x}) >$ if

1) $\overleftarrow{x}_{(1)}$ is not worse than $\overleftarrow{x}_{(2)}$ over all objectives. For example, $f_3(\overleftarrow{x}_{(1)}) \leq f_3(\overleftarrow{x}_{(2)})$ if $f_3(\overleftarrow{x})$ is a minimization objective.
2) $\overleftarrow{x}_{(1)}$ is strictly better than $\overleftarrow{x}_{(2)}$ in at least one objective. For example, $f_6(\overleftarrow{x}_{(1)}) > f_6(\overleftarrow{x}_{(2)})$ if $f_6(\overleftarrow{x})$ is a maximization objective.

Accordingly, the goals of multi-objective optimization can be summarized [1] as: *i)* to find a set of solutions as close as possible to the Pareto-optimal front, and *ii)* to find a set of solutions as diverse as possible. Of course, due to different preference relations, the aim of a decision maker can be different, e.g., finding solutions in a predefined region of the search space etc.

It is natural to use evolutionary algorithms for solving multi-objective optimization problems (MOEA in general and MOGA if based on genetic algorithms), because an evolutionary algorithm constructs a population of individuals, which evolve through time until stopping criteria are satisfied. At any particular time, the current population of individuals represent the current solutions to the input problem, with the final population representing the algorithm's resulting output solutions. An enhancement to the traditional evolutionary algorithm, is to allow an individual to have more than one measure of fitness within a population. In this case, the evolutionary algorithm will have more difficulty selecting

individuals for inclusion in the next population; because a set of individuals contained in one population exhibits a Pareto Front of best individuals (i.e. a set), rather than a single best individual. Several algorithms inspired by this principle have been proposed. Among them, VEGA [3], HLGA [4], NSGA, NSGA-II [5], [6], [7], SPEA [8] and many others.

III. Visualization of High Dimensional Spaces

There are many possible paradigms for creating visual spaces within data mining. In particular Virtual Reality (VR) is a suitable paradigm for several reasons. It is *flexible*: allows the choice of different ways to represent the objects according to differences in human perception. VR allows *immersion*: the user can navigate inside the data and interact with the objects in the world. It creates a *living* experience: the user is a passive observer, but an actor in the world. VR is *broad and deep*: the user may see the VR world as a whole, or concentrate on details. Also very important is that its use does not require any special background knowledge. A virtual reality technique for visual data mining on heterogeneous, imprecise and incomplete information systems was introduced in [9], [10] (see also `http://www.hybridstrategies.com`). It is oriented to the understanding of large heterogeneous, incomplete and imprecise data, as well as symbolic knowledge. The notion of data is not restricted to databases, but includes logical relations and other forms of both structured and non-structured knowledge.

One of the steps in the construction of a VR space for data representation is the transformation of the original set of attributes describing the objects under study, often defining a heterogeneous high dimensional space, into another space of small dimension (typically 2-3) with intuitive metric (e.g. Euclidean). The operation usually involves a non-linear transformation; implying some information loss. There are basically two kinds of spaces sought: *i)* spaces preserving the structure of the objects as determined by the original set of attributes, and *ii*) spaces preserving the distribution of an existing class defined over the set of objects.

Different information sources are associated with the attributes, relations and functions. They are described by the so called source sets (Ψ_i), constructed according to the nature of the information to represent. Source sets also account for incomplete information. A heterogeneous domain [10] is a Cartesian product of a collection of source sets: $\hat{\mathcal{H}}^n = \Psi_1 \times \cdots \times \Psi_n$, where $n > 0$ is the number of information sources to consider. For example, in a domain where objects are described by continuous crisp quantities, discrete features, fuzzy features, time-series, images, and graphs, they can be represented as Cartesian products of subsets of real numbers($\hat{R}$), nominal ($\hat{N}$) or ordinal sets($\hat{O}$), fuzzy sets($\hat{F}$), sets of images ($\hat{I}$), sets of time series ($\hat{S}$) and sets of graphs ($\hat{G}$), respectively (all extended for allow missing values). The heterogeneous domain is $\hat{\mathcal{H}}^n = \hat{N}^{n_N} \times \hat{O}^{n_O} \times \hat{R}^{n_R} \times \hat{F}^{n_F} \times \hat{I}^{n_I} \times \hat{S}^{n_S} \times \hat{G}^{n_G}$, where n_N is the number of nominal sets, n_O of ordinal sets, n_R of real-valued sets , n_F of fuzzy sets , n_I of image-valued sets, n_S of time-series sets, and n_G of graph-valued sets, respectively ($n = n_N + n_O + n_R + n_F + n_I + n_S + n_G$).

A *virtual reality space* is the tuple $\Upsilon =< \underline{O}, G, B, \Re^m, g_o, l, g_r, b, r >$, where $\underline{O}$ is a relational structure ($\underline{O} =< O, \Gamma^v >$, O is a finite set of objects, and Γ^v is a set of relations); G is a non-empty set of *geometries* representing the different objects and relations; B is a non-empty set of *behaviors* of the objects in the virtual world; $\Re^m \subset \mathbb{R}^m$ is a *metric space* of dimension m (euclidean or not) which will be the actual virtual reality geometric space. The other elements are mappings: $g_o : O \to G$, $\varphi : O \to \Re^m$, $g_r : \Gamma^v \to G$, $b : O \to B$.

IV. Visualizing Objective Spaces

Three general approaches for constructing m-dimensional Pareto-optimal fronts [11] have been previously reported in the literature for the purpose of systematically investigating the properties of multi-objective evolutionary algorithms (MOEAs) [12]. Some design characteristics of the approaches for test problem construction are that they should be: *i)* simple to implement, *ii)* scalable to any number of decision variables and objectives and *iii)* lead to knowledge of the exact shape and location of the resulting Pareto-optimal front. Clearly, other characteristics are also of interest from the point of view of constructing test problems that are difficult for MOEAs to solve. However, once such a front has been constructed, either theoretically [11] or empirically via a particular MOEA, it becomes of interest to investigate the properties of a given objective space.

Data structure is one of the most important elements to consider and this is the case when the location and adjacency relationships between the objects O in Υ should give an indication of the *similarity relationships* [13], [14] between the objects in $\hat{\mathcal{H}}^n$ (in the classical cases studied in MOO $\hat{\mathcal{H}}^n = \Re^n$), as given by the set of original attributes [10]. φ can be constructed to maximize some metric/non-metric structure preservation criteria as has been done for decades in multidimensional scaling [15], [14], or to minimize some error measure of information loss [16]. If δ_{ij} is a dissimilarity measure between any two objects i, j ($i, j \in [1, N]$, where N is the number of objects), and $\zeta_{i^v j^v}$ is another dissimilarity measure defined on objects $i^v, j^v \in O$ from Υ ($i^v = \phi(i), j^v = \phi(j)$). Examples of error measures frequently used are [15], [16] and [14]:

$$\text{S stress} = \sqrt{\frac{\sum_{i<j} (\delta_{ij}^2 - \zeta_{ij}^2)^2}{\sum_{i<j} \delta_{ij}^4}}, \quad (1)$$

$$\text{Sammon error} = \frac{1}{\sum_{i<j} \delta_{ij}} \frac{\sum_{i<j} (\delta_{ij} - \zeta_{ij})^2}{\delta_{ij}} \quad (2)$$

$$\text{Quadratic Loss} = \sum_{i<j} (\delta_{ij} - \zeta_{ij})^2 \quad (3)$$

Classical algorithms have been used for directly optimizing these measures, like Steepest descent, conjugate gradient Fletcher-Reeves, Powell, Levenberg-Marquardt, and others. The number of different similarity, dissimilarity and distance

functions definable for the different kinds of source sets is immense. Moreover, similarities and distances can be transformed into dissimilarities according to a wide variety of schemes, thus providing a rich framework.

From the point of view of the property(s) which the objects in the space must satisfy, several paradigms can be considered for building a transformed space for constructing visual representations [17]:

- *Unsupervised*: The location of the objects in the space should preserve some structural property of the data, dependent only on the set of descriptor attributes. Any class information is ignored. The space sought should have minimal distortion.
- *Supervised*: The goal is to produce a space where the objects are maximally discriminated w.r.t. a class distribution. The preservation of any structural property of the data is ignored, and the space can be distorted as much as required in order to maximize class discrimination.
- *Mixed*: A space compromising the two goals is sought. Some amount of distortion is allowed in order to exhibit class differentiation and the object distribution should retain in a degree the structural property defined by the descriptor attributes. Very often these two goals are conflicting.

From the point of view of their mathematical nature, the mappings can be:

- *Implicit*: the images of the transformed objects are computed directly and the algorithm does not provide a function representation.
- *Explicit*: the function performing the mapping is found by the procedure and the images of the objects are obtained by applying the function. Two sub-types are:
  - *analytical functions*: for example, as an algebraic representation constructed via genetic programming.
  - *general function approximators*: for example, as neural networks, fuzzy systems, or others.

An explicit φ is useful for both practical and theoretical reasons. On one hand, in dynamic data sets (e.g. systems being monitored or incremental data bases) an explicit transform φ will speed up the update of the VR information system. On another hand, it can give semantics to the attributes of the VR space, thus acting as a general dimensionality reducer.

A. A Theoretical Case: Test Problem DTLZ2

The need of a set of test problems with known and controlled difficulty measures with which MOEA algorithms could be studied systematically has been addressed extensively [12]. Following a parsimonious principle, simple Pareto fronts with known theoretical properties were selected for constructing visual spaces. Test problem DTLZ2 was chosen for the study, whose fronts are segments of spherical surfaces with a straightforward generalization towards higher dimensional objective spaces (hyperspherical surfaces) (Eq.4). A collection of 5 fronts with the following radii $r = \{1, 5, 10, 20, 50\}$ in objective spaces of dimension $M = \{3, 4, 5, 7, 10\}$ were generated. In all cases the dimension of the decision space was kept the same as the corresponding objective space. For each decision space a mesh of sampling points was generated so that each decision variable $x_i \in [1, M-1]$ was sampled from a regular interval in the $[0, 1]$ domain. If n_j is the number of sampling points along a given decision variable in a j-dimensional decision space, then for each dimension determined by $1/n_j$ the number of points in the theoretical front is n_j^{M-1}. In the study $j \in \{3, 4, 5, 7, 10\}$ and $n_j = \{5, 5, 3, 3, 3\}$. For $M = 3$ the sequence of theoretical Pareto fronts is shown in Fig.1. The chosen design allows: *i)* the simulation of a MO-process "converging" towards the first front of the sequence (lower right) and *ii)* the reproduction of the same type of relationship in dimensions larger than 3 which makes the interpretation of the obtained VR spaces easier.

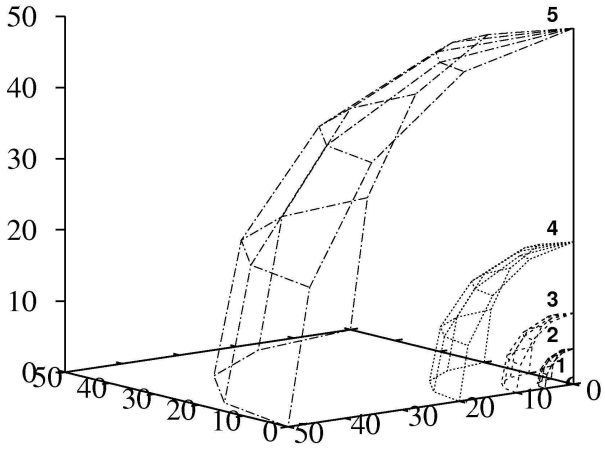


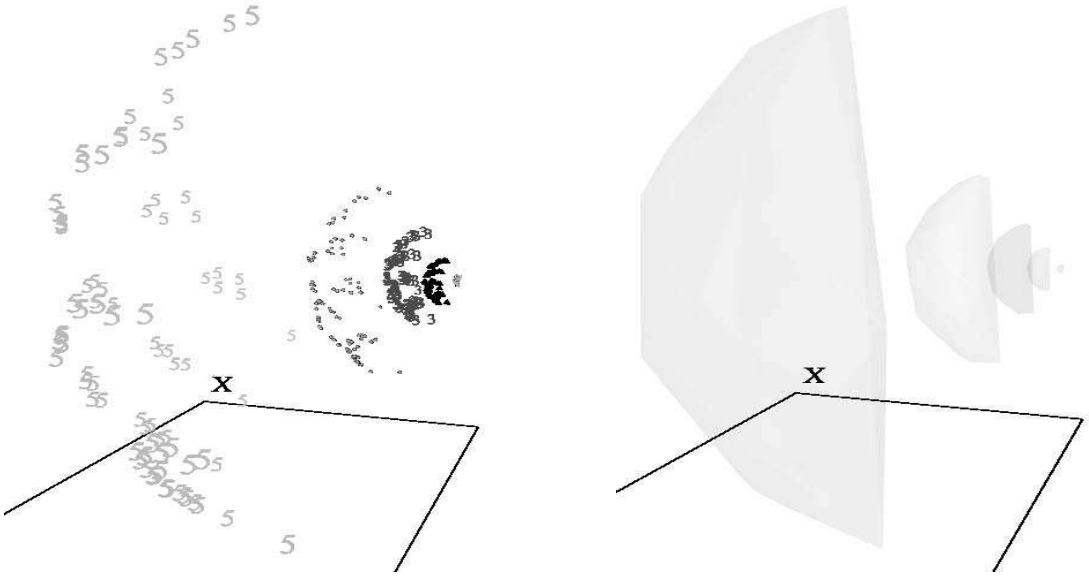
Fig. 1. An example of five theoretical 3-D Pareto fronts with radii 1, 5, 10, 20 and 50 respectively. They correspond to fronts generated from test problem DTLZ2.

B. An Applied Case: The Knapsack problem

A 0/1 knapsack problem [18], [19], [20] is a real world situation that consists of *i)* a set of items, *ii)* weight and profit associated with each item, and *iii)* an upper bound for the capacity of the knapsack. The task is to find a subset of items which will be placed within the knapsack, which maximizes the total profit in the subset (the total weight must not exceed the given capacity of the knapsack). This problem is extended to the multi-objective case by allowing any given number of knapsacks.

This problem is included in the Test Problem Suite for Multiobjective Optimizers [21]. In particular, data and runs using a collection of different MOGA algorithms are compiled for the 750 items, 4 knapsacks case. From them, the results corresponding to objective spaces generated with the HLGA [4], NSGA [5] and VEGA [3] algorithms for optimization run 1 were used in this study (files HLGA.1, NSGA.1 and VEGA.1). A broad comparative study including

$$
\begin{aligned}
\text{Minimize} \quad & f_1(\mathbf{x}) = (1 + g(\mathbf{x}_M))cos(x_1\pi/2)cos(x_2\pi/2)\cdots cos(x_{M-2}\pi/2)cos(x_{M-1}\pi/2) \\
\text{Minimize} \quad & f_2(\mathbf{x}) = (1 + g(\mathbf{x}_M))cos(x_1\pi/2)cos(x_2\pi/2)\cdots cos(x_{M-2}\pi/2)sin(x_{M-1}\pi/2) \\
\text{Minimize} \quad & f_3(\mathbf{x}) = (1 + g(\mathbf{x}_M))cos(x_1\pi/2)cos(x_2\pi/2)\cdots sin(x_{M-2}\pi/2) \\
\vdots \quad & \vdots \\
\text{Minimize} \quad & f_{M-1}(\mathbf{x}) = (1 + g(\mathbf{x}_M))cos(x_1\pi/2)sin(x_2\pi/2) \\
\text{Minimize} \quad & f_{M-1}(\mathbf{x}) = (1 + g(\mathbf{x}_M))sin(x_1\pi/2) \\
& 0 \leq x_i \leq 1, \text{for } i = 1, \ldots, n \\
& g(\mathbf{x}_M) = \sum_{x_i \in \mathbf{x}_M} (x_i - 0.5)^2
\end{aligned}
\tag{4}
$$



(a) 3D representation showing mapped solution locations from a 4D objective space. Geometries: "5" = radius 50, "ball" = radius 20, "3" = radius 10, "cone" = radius 5 and "rod" = radius 1. Behavior = static.

(b) Leftmost Convex Hull wraps the largest radius (50) front solutions with successive decreasing fronts until the rightmost hull, which wraps the smallest radius (1) front solutions. Geometry = Convex Hulls, Behavior = static.

Fig. 2. 4D Test Problem DTLZ2 mapped down to 3D (Sammon Error: 0.003289) for fronts constructed with radii 1, 5, 10, 20 and 50 respectively.

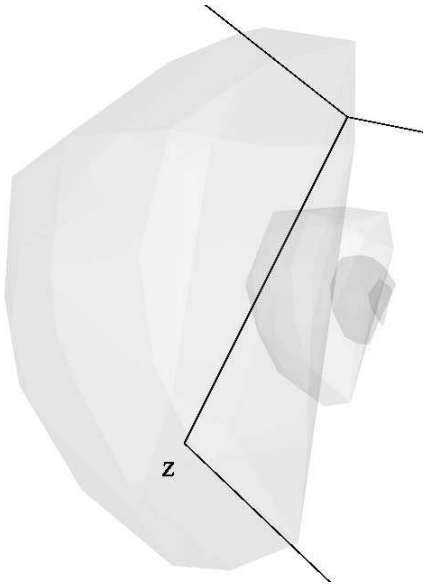
these and other algorithms has been made [8]. The 4-dimensional objective spaces represent approximations to the Pareto front obtained with the algorithms mentioned and they were represented as VR 3-D spaces using the approach described here. The idea was to study the behavior of the proposed MO visualization technique using non-theoretical objective spaces. In this way, a visual comparison between the results given by the HLGA, NSGA and VEGA is possible, complementing the numeric comparison already made [8].

V. Results

For simplicity, all VR spaces were computed from Eq.2 using Euclidean distance in both spaces as the dissimilarity measure. The number of points grows exponentially with the dimension of the decision space; therefore a subset of them were used in the computation. The leader clustering algorithm [22] was used for extracting a kernel set ensuring that no similarity structure was lost beyond a specified threshold. Table II shows the thresholds and the resulting number of points for each problem. Gower's coefficient [23] was used as a similarity measure for the leader algorithm because of its simplicity and insensitivity to scaling differences among the decision variables.

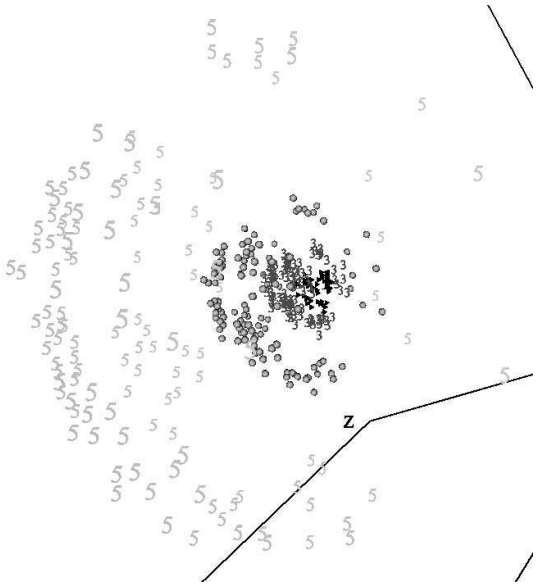
An implicit mapping for Eq.2 was computed using the classical Fletcher-Reeves conjugate gradient algorithm and for each problem, the best solution from 100 random initial configurations was obtained. Many different optimization procedures are possible (classical, evolutionary, hybrid) when constructing solutions for Eq.2. Table I shows the mapping errors. In general they are small, although for $M = 10$ the error is much larger in comparison with those of lower dimension.

(a) 3D representation showing mapped solution locations from a 5D objective space. Geometries: “5” = radius 50, “ball” = radius 20, “3” = radius 10, “2” = radius 5 and “1” = radius 1. Behavior = static.

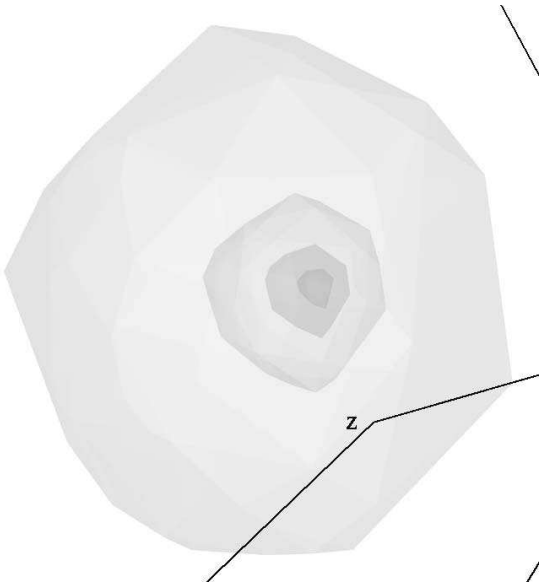


(b) Leftmost Convex Hull wraps the largest radius (50) front solutions with successive decreasing fronts until the rightmost hull, which wraps the smallest radius (1) front solutions. Geometry = Convex Hulls, Behavior = static.

Fig. 3. 5D Test Problem DTLZ2 mapped down to 3D (Sammon Error: 0.022153) for fronts constructed with radii 1, 5, 10, 20 and 50 respectively.

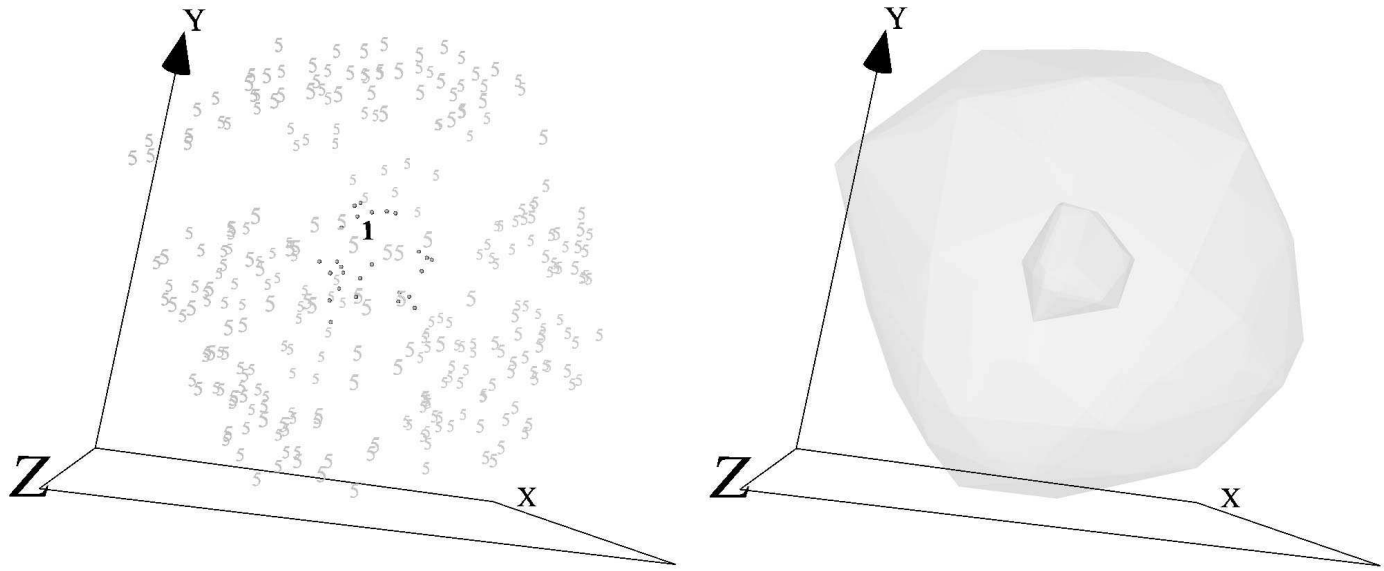


(a) 3D representation showing mapped solution locations from a 7D objective space. Geometries: “5” = radius 50, “ball” = radius 20, “3” = radius 10, “2” = radius 5 and “1” = radius 1. Behavior = static.



(b) Outermost Convex Hull wraps the largest radius (50) front solutions with successive decreasing fronts until the innermost hull, which wraps the smallest radius (1) front solutions. Geometry = Convex Hulls, Behavior = static.

Fig. 4. 7D Test Problem DTLZ2 mapped down to 3D (Sammon Error: 0.023354) for fronts constructed with radii 1, 5, 10, 20 and 50 respectively.



(a) 3D representation showing mapped solution locations from a 10D objective space. Geometries: "5" = radius 50, "ball" = radius 20, "bold 3" = radius 10, and "bold 1" = radius 1. Behavior = static.

(b) Largest Convex Hull wraps the largest radius (50) front solutions followed by radius (20) and (10) front solutions. The fourth convex hull (not shown) degenerates to a point for radius (1) front solutions.

Fig. 5. 10D Test Problem DTLZ2 mapped down to 3D (Sammon Error: 0.056397) for fronts constructed with radii 1, 5, 10, 20 and 50 respectively.

TABLE I
RESULTS FOR THE COMPUTATION OF THE NEW SPACE

Problem Dimension	Sammon Error	Relative Error	Number of Iterations
Theoretical Fronts			
4	0.003289	0.000010	231
5	0.022153	0.000008	104
7	0.023354	0.000009	127
10	0.056397	0.000008	80
Knapsack Problem			
4	0.011820	$1.0e^{-9}$	5

TABLE II
RESULTS FOR THE LEADER ALGORITHM

Dimension	Number of Objects	Similarity Threshold	Number of Leaders
Theoretical Fronts			
5	3125	0.98	615
7	3645	0.987	398
10	98415	0.94	309
Knapsack Problem			
HLGA	225	0.945	123
NSGA	314	0.945	126
VEGA	196	0.945	96

A. Results for Test Problem DTLZ2

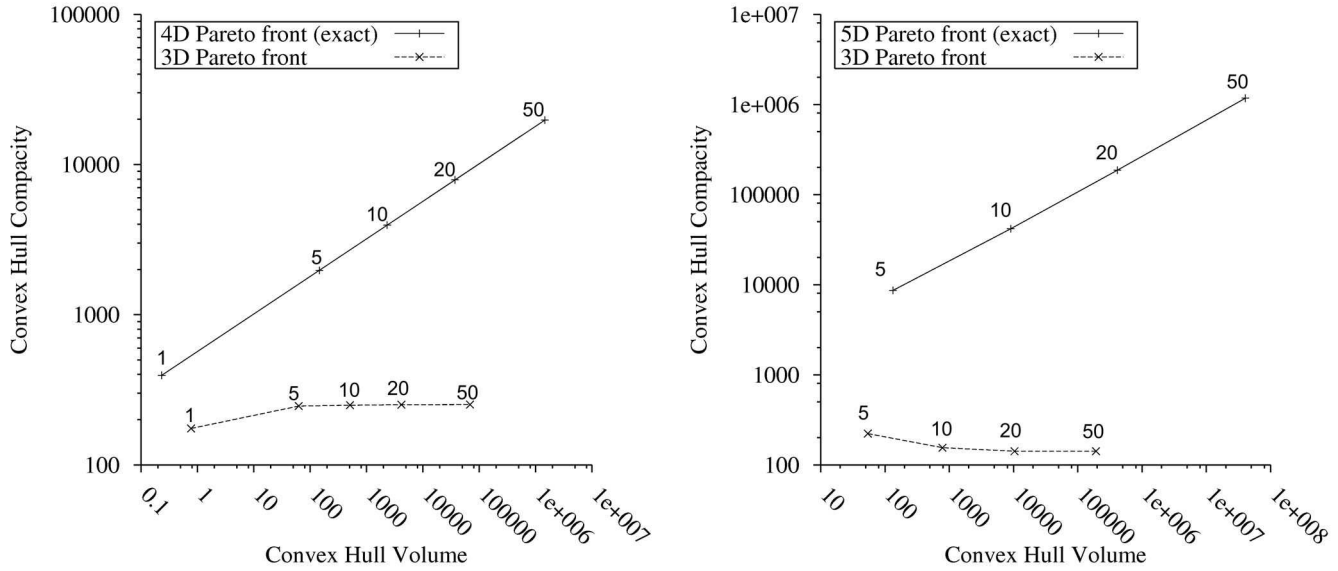
The 3D representation of the sequence of 4D fronts is shown in Fig. 2(a). Their shape is that of "thick" curved surfaces and their progression in size and ordering follows the same relationship as the pure 3D case of Fig. 1. This situation is also clear in Fig. 2(b) (e.g. showing the progression of the convex hulls). The accuracy of this representation is high as indicated by the low Sammon error in Table I.

For 5D the situation is shown in Figs. 3(a), 3(b). Although the information loss is much larger than for 4D, the behavior of the mapped fronts is similar to the 4D and the original 3D cases. For 7D (Figs. 4(a), 4(b)) the fronts are still curved surfaces in 3D and their embedding keeps the same relation as the pure 3D. The error is very similar to that of the 5D case. The convex hulls are more spherical, but still asymmetric, having the radius $= 50$ front as outer shell and the radius $= 1$ as core. In 10 D (Figs. 5(a), 5(b)) the main features of the original 3D case are still recognizable (proper sequencing, embedding and asymmetry), although the surfaces are more deformed due to the much larger Sammon error they still exhibit the features of the lower dimensional mapping (e.g. 7 D).

For comparison, Figs. 6(a), 6(b), and 7(a) show the relation between two simple shape measures, the convex hull volume H_v and its compacity H_c [24], [25] as given by $H_c = (area)^3/(volume)^2$. In the original high dimensional spaces the relationship between H_c and H_v clearly follows a power law. For the mapped cases, this property is lost although the relative ordering of the Pareto fronts is preserved. One measure of quality improvement for forthcoming algorithms could be the proximity to the theoretical upper limit.

B. Results for The Knapsack problem

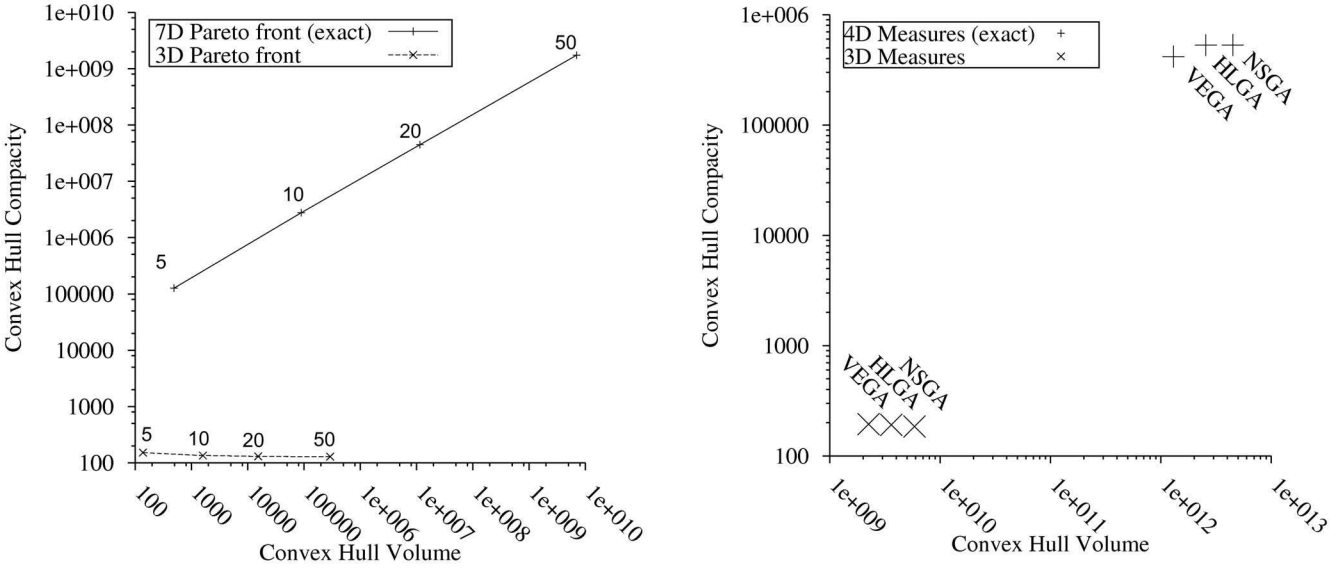
The results obtained from [8] for the 4 knapsacks problem with 750 items are visually presented in Fig. 8 as a 3 dimensional space where point locations (knapsack solutions) are shown for each of the 3 fronts separately obtained from 3 different multi-objective algorithms. The curved Pareto front is visible along with the relative performance of the 3 algorithm's solution populations, which are in agreement with



(a) Effect of increasing radii on theoretical 4D and mapped 3D fronts for radii 1, 5, 10, 20 and 50. Each 3D mapping is the best result from 100 trials.

(b) Effect of increasing radii on theoretical 5D and mapped 3D fronts for radii 5, 10, 20 and 50. Each 3D mapping is the best result from 100 trials.

Fig. 6. Theoretical and mapped results (measures of the convex hulls) for the 4D and 5D test problem DTLZ2.



(a) Effect of increasing radii on theoretical 7D and mapped 3D fronts for radii 5, 10, 20 and 50. Each 3D mapping is the best result from 100 trials.

(b) Comparison of results from 3 algorithms (HLGA, NSGA, and VEGA) using measures computed on both the original 4D convex hulls and mapped 3D convex hulls, where each convex hull wraps the solutions obtained by the respective algorithm.

Fig. 7. Theoretical and mapped results (measures of the convex hulls) for the 7D theoretical test problem DTLZ2 and 4D applied (as computed by 3 algorithms) knapsack problem.

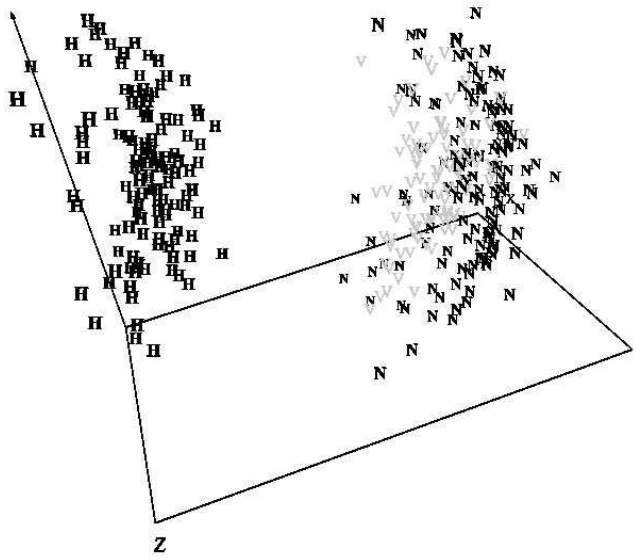


Fig. 8. 4 Knapsacks Problem with 750 items. The 4D objective space has been mapped down to 3D (Sammon Error: 0.011820). (H: HLGA algorithm solutions, N: NSGA algorithm solutions, and V: VEGA algorithm solutions) Geometry = letters, Behavior = static.

previously reported results [8]. The static Fig. 8 shows that the coarse domination order is NSGA followed by VEGA then HLGA with Fig. 7(b) showing additional characteristics of the respective solution sets.

VI. Conclusions

Multi-objective problems lead to constructing multiple solutions along a Pareto front. Such solutions lie in a high dimensional objective space. An approach for visualizing one or more sets of high dimensional fronts has been proposed. Results were obtained for both examples of the *i)* theoretical fronts derived from test problem DTLZ2 in $3, 4, 5, 7$ and 10 dimensions and *ii)* approximations to Pareto fronts for the 4 dimensional knapsack problem obtained by known MOGA approaches (HLGA, NSGA, and VEGA). The VR spaces of their high dimensional counterparts show clearly the expected characteristics of the fronts in terms of relative sizes, sequencing, embedding and asymmetry. In addition, the characteristics of the mapped solutions were shown to leave room for improvement for future proposed algorithms. These are preliminary results for which further experimentation would be required and they show a potential for aiding decision making and algorithm development in multi-objective optimization problems.

Acknowledgment

The authors would like to thank Robert Orchard from the Integrated Reasoning Group (National Research Council Canada, Institute for Information Technology) for reading a first draft of this paper.

References

[1] K. Deb, *Multi-Objective Optimization using Evolutionary Algorithms*. John Wiley and Sons, 2001.

[2] E. K. Burke and G. Kendall, *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques*. 233 Spring Street, New York, NY 10013, USA: Springer Science and Business Media, Inc., 2005, no. 0-387-23460-8.

[3] R. Schaffer, "Multiple objective optimization with vector evaluated genetic algorithms," in *Proc. First International Conference on Genetic Algorithms*, 1985, pp. 93–100.

[4] P. Hajela and C. Lin, "Genetic search strategies in multicriterion optimal design," *Structural Optimization*, vol. 4, pp. 99–107, 1992.

[5] N. Srinivas and K. Deb, "Multiobjective optimization using nondominated sorting in genetic algorithms," *Evol. Comput.*, vol. 2, no. 3, pp. 221–248, 1994.

[6] K. Deb, S. Agarwal, A. Pratap, and T. Meyarivan, "A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: Nsga-ii," in *Proceedings of the Parallel Problem Solving from Nature VI Conference*, Paris, France, 16-20 September 2000, pp. 849–858.

[7] K. Deb, S. Agarwal, and T. Meyarivan, "A fast and elitist multi-objective genetic algorithm: Nsga-ii." in *IEEE Transaction on Evolutionary Computation*, vol. 6 (2), 2002, pp. 181–197.

[8] E. Zitzler and L. Thiele, "Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 4, pp. 257–271, 1999.

[9] J. J. Valdés, "Virtual reality representation of relational systems and decision rules:," in *Theory and Application of Relational Structures as Knowledge Instruments*, P. Hajek, Ed. Prague: Meeting of the COST Action 274, Nov 2002.

[10] ——, "Virtual reality representation of information systems and decision rules:," in *Lecture Notes in Artificial Intelligence*, ser. LNAI, vol. 2639. Springer-Verlag, 2003, pp. 615–618.

[11] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable Test Problems for Evolutionary Multi-Objective Optimization," Kanpur Genetic Algorithms Laboratory (KanGAL), Department of Mechanical Engineering, Indian Institute of Technology, Kanpur, Tech. Rep. KanGAL Report Number 2001001, 2001.

[12] K. Deb, "Multi-objective genetic algorithms: Problem difficulties and construction of test problems," *Evolutionary Computation*, vol. 7, no. 3, pp. 205–230, 1999.

[13] J. L. Chandon and S. Pinson, *Analyse typologique. Théorie et applications*. Masson, Paris, 1981.

[14] I. Borg and J. Lingoes, *Multidimensional similarity structure analysis*. Springer-Verlag, 1987.

[15] J. Kruskal, "Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis," *Psichometrika*, vol. 29, pp. 1–27, 1964.

[16] J. W. Sammon, "A non-linear mapping for data structure analysis," *IEEE Trans. Computers*, vol. C18, pp. 401–408, 1969.

[17] J. J. Valdés and A. J. Barton, "Virtual reality spaces for visual data mining with multiobjective evolutionary optimization: Implicit and explicit function representations mixing unsupervised and supervised properties," in *IEEE Congress of Evolutionary Computation (CEC 2006)*. Vancouver: IEEE, July 16-21 2006, pp. 5592–5598.

[18] S. Martello and P. Toth, *Knapsack Problems: Algorithms and Computer Implementations*. John Wiley and Sons, 1990.

[19] S. Khuri and T. Bäck and J. Heitkötter, "The zero/one multiple knapsack problem and genetic algorithms," in *Proc. 1994 ACM Symp. Applied Computing (E. Deaton, D. Oppenheim, J. Urban, and H. Berghel, Eds)*. New York: ACM-Press, 1994, pp. 188–193.

[20] R. Spillman, "Solving large knapsack problems with a genetic algorithm," in *IEEE Int. Conf. Systems, Man and Cybernetics*. Piscataway: IEEE, Oct. 22-25 1995, pp. 632–637.

[21] E. Zitzler and M. Laumanns, "Test problems and test data for multiobjective optimizers," 2006. [Online]. Available: http://www.tik.ee.ethz.ch/~zitzler/testdata.html

[22] J. A. Hartigan, *Clustering Algorithms*. New York: John Wiley & Sons, Inc., 1975.

[23] J. C. Gower, "A general coefficient of similarity and some of its properties," *Biometrics*, vol. 1, no. 27, pp. 857–871, 1973.

[24] E. Bribiesca, "A measure of compactness for 3d shapes," *Computers and Mathematics with Applications*, vol. 40, pp. 1275–1284, 2000.

[25] J. Corney, H. Rea, D. Clark, J. Pritchard, M. Breaks, and R. Macleod, "Coarse filters for shape matching," *Computer Graphics and Applications*, vol. 22, no. 3, pp. 65–74, 2002.