Computing the Set of Approximate Solutions of a Multi-objective Optimization Problem by Means of Cell Mapping Techniques

Carlos Hernández[1], Jian-Qiao Sun[2], and Oliver Schütze[1]

[1] CINVESTAV-IPN,
Computer Science Department
Av. IPN 2508, Col. San Pedro Zacatenco
07360 Mexico City, Mexico
chernandez@computacion.cs.cinvestav.mx,
schuetze@cs.cinvestav.mx

[2] University of California Merced
School of Engineering
Merced, CA 95344, USA
jsun3@ucmerced.edu

Abstract. Here we address the problem of computing the set of approximate solutions of a given multi-objective optimization problem (MOP). This set is of potential interest for the decision maker since it might give him/her additional solutions to the optimal ones for the realization of the project related to the MOP. In this study, we make a first attempt to adapt well-known cell mapping techniques for the global analysis of dynamical systems to the problem at hand. Due to their global approach, these methods are well-suited for the thorough investigation of small problems, including the computation of the set of approximate solutions. We conclude this work with the presentation of three academic bi-objective optimization problems including a comparison to a related evolutionary approach.

Keywords: multi-objective optimization, approximate solutions, cell mapping techniques, global optimization.

1 Introduction

In many real-world engineering problems one is faced with the problem that several objectives have to be optimized concurrently leading to a multi-objective optimization problem (MOP). Such problems can be stated as

$$\min_x F : Q \subset \mathbb{R}^n \to \mathbb{R}^k. \tag{1}$$

The solution set of a MOP, the Pareto set P_Q, typically forms a $(k-1)$-dimensional entity, where k is the number of objectives involved in the problem. Hence, the approximation of P_Q already represents a challenge for many search procedures, in particular if the objectives are nonlinear. However, in some applications it might be interesting to know in addition to the optimal solutions

M. Emmerich et al. (eds.), *EVOLVE - A Bridge between Probability, Set Oriented Numerics, and Evolutionary Computation IV*, Advances in Intelligent Systems and Computing 227,
DOI: 10.1007/978-3-319-01128-8_12, © Springer International Publishing Switzerland 2013

also nearly optimal ones. The reason is that if for two vectors $x, y \in Q$ the objective values $F(x)$ and $F(y)$ are close, this does not have to hold for x and y. Hence, if the decision maker (DM) is willing to accept a certain small deterioration ϵ in the performance (which is measured by F), he/she might be given several backup alternatives to realize the project. The approximation of the set of nearly optimal solutions $P_{Q,\epsilon}$ has caught little attention in literature so far. The main reason for this lack might be that this set is n-dimensional, where n is the dimension of the decision space.

In this work, we propose to use cell mapping techniques ([20]) for the approximation of $P_{Q,\epsilon}$. Methods of that kind divide the domain Q into a set of small n-dimensional cells and perform a cell-to-cell mapping of the given dynamical system g (here we will use a particular system derived from a Pareto descent method). In this way, a global view on the dynamics of g as well as the fitness landscape of F on Q are obtained, and is thus well-suited for the problem at hand. The result of the algorithm is ideally a tight covering of $P_{Q,\epsilon}$, but also the discretization of this set is straightforward (e.g., one representative of each cell can be chosen) which is required for the presentation of the result to the DM. Due to its approach, the method is restricted to small dimensions of the parameter space. However, there exist small dimensional problems where a thorough investigation is desirable. Such models e.g. arise in preliminary space mission design (e.g., [12,36,35]) or in the design of electrical circuits (e.g., [3,2]).

In literature, there exists a huge variety of methods for the approximation of the Pareto set P_Q (respectively its image, the Pareto front). There exist, for instance, point-wise iterative mathematical programming techniques that lead to single solutions of a MOP. An approximation of P_Q can be obtained by choosing a clever sequence of these problems (e.g., [8,23,15,14]). Further, there exist set oriented methods such as multi-objective evolutionary algorithms (MOEAs, see e.g., [9,5]) or subdivision techniques ([13,22,37]) that aim for the approximation of P_Q in one run of the algorithm. None of them, however, are designed for the computation of $P_{Q,\epsilon}$. The only studies in that direction seem to be the works [32,35], where stochastic search techniques such as MOEAs are investigated.

The remainder of this paper is organized as follows: In Section 2, we present the background required for the understanding of the sequel. In Section 3, we propose a cell mapping technique for the computation of the set of approximate solutions. In Section 4, we present some numerical results, and the final Section states the conclusions and future work.

2 Background

Here we will shortly recall the required background: The concept of multi-objective optimization including our definition of nearly optimality and a brief review of cell mapping techniques.

A multi-objective optimization problem (MOP) can be expressed as follows:

$$\min_{x \in Q} \{F(x)\}, \tag{MOP}$$

where F is the map that consists of the objective functions $f_i : Q \to \mathbb{R}$ under consideration, i.e.,

$$F : Q \to \mathbb{R}^k, \qquad F(x) = (f_1(x), \dots, f_k(x)).$$

The domain $Q \subset \mathbb{R}^n$ of F can in general be expressed by inequality and equality constraints:

$$Q = \{x \in \mathbb{R}^n \mid g_i(x) \leq 0,\ i = 1, \dots, l, \text{ and } h_j(x) = 0,\ j = 1, \dots, m\}.$$

In this work we will merely consider inequality constraints.

Next, we have to define optimal solutions of a given MOP. This can e.g. be done using the concept of *dominance* ([27]).

Definition 1. *(a) Let $v, w \in \mathbb{R}^k$. Then the vector v is* less than *w (in short: $v <_p w$), if $v_i < w_i$ for all $i \in \{1, \dots, k\}$. The relation $\leq_p$ is defined analogously.*

(b) A vector $y \in Q$ is called dominated *by a vector $x \in Q$ ($x \prec y$) with respect to (MOP) if $F(x) \leq_p F(y)$ and $F(x) \neq F(y)$, else y is called non-dominated by x.*

If a vector x dominates a vector y, then x can be considered to be 'better' according to the given MOP. The definition of optimality (i.e., of a 'best' solution) of a given MOP is now straightforward.

Definition 2. *(a) A point $x \in Q$ is called* (Pareto) optimal *or a* Pareto point *of (MOP) if there is no $y \in Q$ that dominates x.*

(b) The set of all Pareto optimal solutions is called the Pareto set*, i.e.,*

$$\mathcal{P} := \{x \in Q\ :\ x \text{ is a Pareto point of } (MOP)\}. \tag{2}$$

(c) The image $F(\mathcal{P})$ of $\mathcal{P}$ is called the Pareto front.

Pareto set and Pareto front typically form $(k-1)$-dimensional objects under certain mild assumptions on the MOP, see [17] for a thorough discussion.

We now define another notion of dominance which we use to define approximate solutions.

Definition 3 ([31]). *Let $\epsilon = (\epsilon_1, \dots, \epsilon_k) \in \mathbb{R}^k_+$ and $x, y \in Q$.*

(a) x is said to ϵ-dominate y ($x \prec_\epsilon y$) with respect to (MOP) if $F(x) - \epsilon \leq_p F(y)$ and $F(x) - \epsilon \neq F(y)$.

(b) x is said to $-\epsilon$-dominate y ($x \prec_{-\epsilon} y$) with respect to (MOP) if $F(x) + \epsilon \leq_p F(y)$ and $F(x) + \epsilon \neq F(y)$.

The notion of $-\epsilon$-dominance is of course analogous to the 'classical' ϵ-dominance relation [24] but with a value $\tilde{\epsilon} \in \mathbb{R}^k_-$. However, we highlight it here since we use it to define our set of interest:

Definition 4. *Denote by $P_{Q,\epsilon}$ the set of points in $Q \subset \mathbb{R}^n$ that are not $-\epsilon$-dominated by any other point in Q, i.e.,*

$$P_{Q,\epsilon} := \{x \in Q \mid\ \nexists y \in Q : y \prec_{-\epsilon} x\}. \tag{3}$$

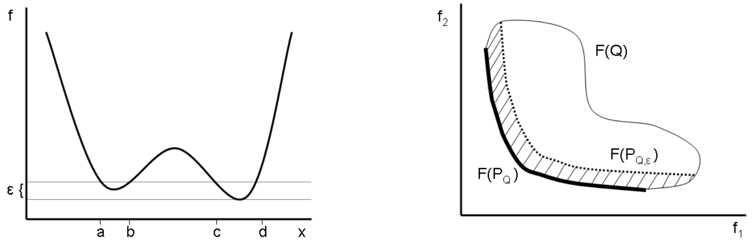


Fig. 1. Two different examples for sets $P_{Q,\epsilon}$. At the left, we show the case for $k = 1$ and in parameter space with $P_{Q,\epsilon} = [a, b] \cup [c, d]$. Note that the image solutions $f([a, b])$ are nearly optimal (measured in objective space), but that the entire interval $[a, b]$ is not 'near' to the optimal solution which is located within $[c, d]$. At the right, we show an example for $k = 2$ in image space, $F(P_{Q,\epsilon})$ is the approximate Pareto front (taken from [31]).

The set $P_{Q,\epsilon}$ contains all ϵ-efficient solutions, i.e., solutions which are optimal up to a given (small) value of ϵ. Figure 1 gives two examples.

To compare our results we will measure the distance of the outcome sets to the set of interest. A natural choice would be the Hausdorff distance d_H (e.g., [16]). Since d_H punishes single outliers that can occur when using stochastic search algorithms (as we will do in our comparison) we will use the averaged Hausdorff distance[1] instead.

Definition 5 ([34]). *Let $p \in \mathbb{N}$, $A = \{a_1, \ldots, a_r\}$ and $B = \{a_1, \ldots, a_m\}$ be two finite sets. Then it is*

$$\Delta_p(A, B) = \max\left(\left(\frac{1}{r}\sum_{i=1}^{r} dist(a_i, B)^p\right)^{1/p}, \left(\frac{1}{m}\sum_{i=1}^{m} dist(b_i, A)^p\right)^{1/p}\right), \quad (4)$$

where $dist(x, B) := \min_{b \in B} \|x - b\|$ denotes the distance between a point x and a set B.

Note that for $p = \infty$, we have $\Delta_\infty = d_H$, and for finite values of p the distances in Δ_p are averaged. In [34], Δ_p is discussed as a performance indicator in the context of Pareto front approximations. In that case, the indicator can be viewed as a combination of slight variations of the well-known indicators *Generational Distance* (see [38]) and the *Inverted Generational Distance* (see [6]).

The cell mapping method was originally proposed by Hsu [18,20] for global analysis of nonlinear dynamical systems in the *state space.* Two cell mapping methods have been extensively studied, namely, the simple cell mapping and the generalized cell mapping. The cell mapping methods have been applied to optimal control problems of deterministic and stochastic dynamic systems [19,4,7].

[1] The averaged Hausdorff is in general not a distance.

The cell mapping methods transform the point-to-point dynamics into a cell-to-cell mapping by discretizing both phase space and the integration time. The simple cell mapping (SCM) offers an effective approach to investigate global response properties of the system. The cell mapping with a finite number of cells in the computational domain will eventually lead to closed groups of cells of the period equal to the number of cells in the group. The periodic cells represent approximate invariant sets, which can be periodic motion and stable attractors of the system. The rest of the cells form the domains of attraction of the invariant sets. For more discussions on the cell mapping methods, their properties and computational algorithms, the reader is referred to the book by Hsu [20].

3 A Cell Mapping Method for the Approximation of $P_{Q,\epsilon}$

In this section, we review the SCM method [20] together with our adaptions to the context of multi-objective optimization, and present the post-processing to get an approximation of $P_{Q,\epsilon}$.

In the following, we assume the problem is bounded by box constraints, which constitutes our domain Q

$$lb_i \leq x_i \leq ub_i,\ i = 1, \cdots, N.$$

Now, we can divide each interval in N_i sections of size

$$h_i = \frac{lb_i - ub_i}{N_i}.$$

By doing this, we get a finite subdivision of the domain, where each of these elements are called regular cells. The number of regular cells is noted by N_c and we label the set of regular cells with positive integers, $1, 2, \cdots, N_c$. Everything that is outside the domain is called the sink cell. With the introduction of the sink cell the total number of cells is $N_c + 1$. The sink cell is also labeled as the regular cells using the value 0.

At this point the evolution of the system is given cell-to-cell instead of point-to-point. The dynamics of a cell z is represented by its center and the cell-to-cell mapping is denoted by C. Now the mapping can be described by

$$\begin{aligned} z(n+1) &= C(z(n)), \quad z(n), z(n+1) \in \{0, \cdots, Nc\}, \\ C(0) &= 0. \end{aligned}$$

A cell z^* that is mapped onto itself,

$$z^* = C(z^*),$$

is called an equilibrium cell.

Definition 6. *A periodic motion of period k for C is a sequence of k cells $z^*(l), l = 1, \cdots, k$, such that*

$$\begin{aligned} z^*(m+1) &= C^m(z^*(1)), \quad m = 1, \cdots, k-1, \\ z^*(1) &= C^k(z^*(1)). \end{aligned}$$

Every element $z^*(l)$ is called a periodic cell of period k or $P - k$ group.

Definition 7. *We say a cell z has distance r from a periodic motion if r is the minimum positive integer such that*

$$C^r(z) = z^*(l),$$

where $z^*(l)$ is one of the $P - k$ cells of the motion. The set of such cells is called the r-step basin of attraction of the motion. If r goes to infinity we obtain the basin of attraction.

The evolution of the system starting with any regular cell z can lead only to one of the following three possible outcomes:

- The cell belongs to a P-Group: z is itself a periodic cell of a periodic motion. The evolution of the system simply leads to a periodic motion.
- The cell maps to the sink cell in r steps: Cell z is mapped into the sink cell in r steps. Then the cell belongs to the r-step domain of attraction of the sink cell.
- The cell maps to a P-group in r steps: Cell z is mapped into a periodic cell of a certain periodic motion in r steps. In this case the cell belongs to the r-step domain of attraction of that periodic motion.

To capture the global properties of a cell, the SCM algorithm uses the following sets:

- Group motion number (Gr): The group number uniquely identifies a periodic motion; it is assigned to every periodic cell of that periodic motion and also to every cell in the domain of attraction. The group numbers, positive integers, can be assigned sequentially.
- Period (Pe): Defines the period of each periodic motion.
- Number of steps to a P-group (St): Used to indicate how many steps it takes to map this cell into a periodic cell.

So far the SCM for general dynamical systems. In order to apply it to the context of multi-objective optimization, we have to define a suitable dynamical system. For this, we have chosen to take models that are derived from *descent directions.* A direction $v \in \mathbb{R}^n$ is called a descent direction at a given point $x_0 \in Q$ if a search in that direction leads to an improvement of all objectives. In other words, there exists a $\bar{t} \in \mathbb{R}_+$ such that $F(x_0 + tv) <_p F(x_0)$ $\forall t \leq \bar{t}$. Descent directions are e.g. proposed in [21,29,25,1]. Since we consider bi-objective optimization problems in this work, we use descent direction from [1] which is given by

Theorem 1 ([1]). *Let $x \in \mathbb{R}^n$, and $f_1, f_2 : \mathbb{R}^n \to \mathbb{R}$ define a two-objective MOP. if $\nabla f_i(x) \neq 0$, for $i = 1, 2$, then the direction*

$$v(x_0) := -\left(\frac{\nabla f_1(x_0)}{||\nabla f_1(x_0)||} + \frac{\nabla f_2(x_0)}{||\nabla f_2(x_0)||} \right), \tag{5}$$

is a descent direction at x_0 of MOP. This descent direction yielded best results in our tests but is restricted to problems with two objectives. Using (5), the dynamical system

$$\dot{x}(t) = v(x(t)) \tag{6}$$

can now be used since it defines a pressure toward the Pareto set/front of the MOP at hand.

In order to be able to store those cells that are candidates to be Pareto optimal (to be more precise, cells that potentially contain a part of the Pareto set), we use a set called cPs. For these cells it holds $St(cell) = 0$ and $Gr(cell) \neq 1$. It is important to notice that because of the dynamical system periodic groups with size greater than 1 should not appear, however, due to discretization errors and too large step sizes periodic groups greater than 1 may be generated (i.e., an oscillation around the Pareto set can occur). Thus, cells that are involved in the current periodic group are also considered to be candidates.

Algorithm 1 shows the key elements of the SCM method for the treatment of MOPs. According to the previous discussion, the algorithm works as follows, until all cells are processed, the value of the group motion indicates the state of the current cell and it also points out the corresponding actions to the cell.

A value of $Gr(cell) = 0$ means, the cell has not been processed, hence the state of the cell changes to under process and then we follow the dynamical system to the next cell.

A value of $Gr(cell) = -1$ means, the cell is under processed, which means we have found a periodic group and we can compute the global properties of the current periodic motion.

A value $Gr(cell) > 0$ means, the cell has already been processed, hence we found a previous periodic motion along with its global properties, which can be used to complete the information of the cells under process.

Figure 2 shows an example of a group motion with $N_c = 10 \times 10$ and $Pe(z) = 12$.

After one run of the SCM algorithm, we have gathered the information on the global dynamics of the system and are hence able to approximate the set of interest in a post-processing step. For the problem at hand, the approximation of $P_{Q,\epsilon}$, we use the archiving technique $ArchiveUpdateTight2$ proposed in [26] as follows: Once the group number of the current periodic motion is discovered, we use Algorithm 2 to compute the set $P_{Q,\epsilon}$. Algorithm 2 updates the archive first with the periodic group of the current periodic motion and continues with the rest of the periodic motion. Once it finds a cell which is not in $P_{Q,\epsilon}$ it stops the procedure. The reason for this can be easily seen. Since each periodic group is a curve of dominated points, once a point $x_j \notin P_{Q,\epsilon}$ the other points would not be either, since by construction these points are dominated by x_j.

Algorithm 1. Simple Cell Mapping for MOPs.

```
Require: $DynamicalSystem, F, ub, lb, N, h, N_c$
Ensure: $z, C, Gr, Pe, St, cPs$
$current\_group \leftarrow 1$
$cPs = \{\}$
$Gr(i) \leftarrow 0, \forall i \in N_c$
for all $pcell \in N_c$ do
    $cell \leftarrow pcell$
    $i \leftarrow 0$
    while $newcell =$ true do
        $x_i \leftarrow$ center point of $cell$
        if $Gr(cell) = 0$ then
            $Gr(cell) \leftarrow -1$
            $p_{i+1} \leftarrow DynamicalSystem(x_i)$
            $ncell \leftarrow$ cell where $p_{i+1}$ is located
            $C(cell) \leftarrow ncell$
            $cell \leftarrow ncell$
            $i \leftarrow i + 1$
        end if
        if $Gr(cell) > 0$ then
            $Gr(C^j(pcell)) \leftarrow Gr(cell), j \leftarrow 0, \cdots, i$
            $Pe(C^j(pcell)) \leftarrow Pe(cell), j \leftarrow 0, \cdots, i$
            $St(C^j(pcell)) \leftarrow St(cell) + i - j, j \leftarrow 0, \cdots, i$
            $cell \leftarrow C(cell)$
            $newcell \leftarrow$ false
        end if
        if $Gr(cell) = -1$ then
            $current\_group \leftarrow current\_group + 1$
            $Gr(C^k(pcell)) \leftarrow current\_group, k \leftarrow 0, \cdots, i$
            $j \leftarrow i^{th}$ value when period appears
            $Pe(C^k(pcell)) \leftarrow i - j, k \leftarrow 0, \cdots, i$
            $St(C^k(pcell)) \leftarrow j - k, k \leftarrow 0, \cdots, j - 1$
            $St(C^k(pcell)) \leftarrow 0, k \leftarrow j, \cdots, i$
            $cPs \leftarrow cPs \cup \{x_k\}, k \leftarrow j, \cdots, i$
            $cell \leftarrow C(cell)$
            $newcell \leftarrow$ false
        end if
    end while
end for
```

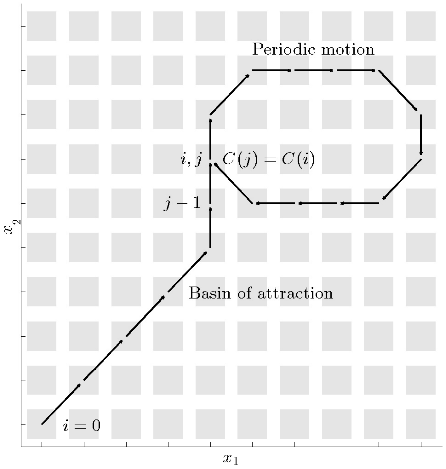


Fig. 2. Example of group motion

Algorithm 2. Post-processing to get $P_{Q,\epsilon}$

```
Require: $P_{Q,\epsilon}, C, pcell, i$
Ensure: $P_{Q,\epsilon}$
j ← i − 1
$isP_{Q,\epsilon}$ ← true
while $isP_{Q,\epsilon}$ do
    $x_j$ ← center point of $C^j(pcell)$
    $P_{Q,\epsilon}$ ← $ArchiveUpdateTight2(P_{Q,\epsilon}, x_j, F(x_j), \epsilon, h)$
    if $x_j \notin P_{Q,\epsilon}$ then
        $isP_{Q,\epsilon}$ ← false
    end if
    $j$ ← $j - 1$
end while
```

4 Numerical Results

Here we present some numerical results on three bi-objective benchmark models. In order to compare the results of the cell mapping technique we couple MOEAs with the archiving technique $ArchiveUpdateP_{Q,\epsilon}$ ([32]) as suggested in [35]. For the MOEAs we have chosen for the state-of-the-art algorithms NSGA-II ([10]) and MOEA/D ([39]).

4.1 Problem 1

First, we consider the MOP taken from [30] that is given by two objective functions $f_1, f_2 : \mathbb{R}^2 \to \mathbb{R}$,

$$\begin{aligned} f_1(x_1,x_2) &= (x_1-1)^2+(x_2-1)^4, \\ f_2(x_1,x_2) &= (x_1+1)^2+(x_2+1)^2. \end{aligned} \tag{7}$$

Here, we have chosen for the domain $Q = [-3,3] \times [-3,3]$. The Pareto set P_Q forms a curve connecting the end points $x_1 = (-1,-1)^T$ and $x_2 = (1,1)^T$. Figure 3 shows the numerical result obtained by the SCM algorithm and the two evolutionary algorithms. Tables 1 and 2 show the Δ_p values of the candidate sets (respectively their images) to $P_{Q,\epsilon}$ ($F(P_{Q,\epsilon})$) for this and all further examples. Apparently, SCM is able to get the best approximation of the set of interest, in particular in parameter space.

4.2 Problem 2

Next, we consider the problem $F : \mathbb{R}^2 \to \mathbb{R}^2$ proposed in [28]

$$F(x_1,x_2) = \begin{pmatrix} (x_1 - t_1(c+2a)+a)^2 + (x_2 - t_2 b)^2 + \delta_t \\ (x_1 - t_1(c+2a)-a)^2 + (x_2 - t_2 b)^2 + \delta_t \end{pmatrix}, \tag{8}$$

where

$$t_1 = \operatorname{sgn}(x_1)\min\left(\left\lceil \frac{|x_1| - a - c/2}{2a+c} \right\rceil, 1\right), t_2 = \operatorname{sgn}(x_2)\min\left(\left\lceil \frac{|x_2| - b/2}{b} \right\rceil, 1\right),$$

and

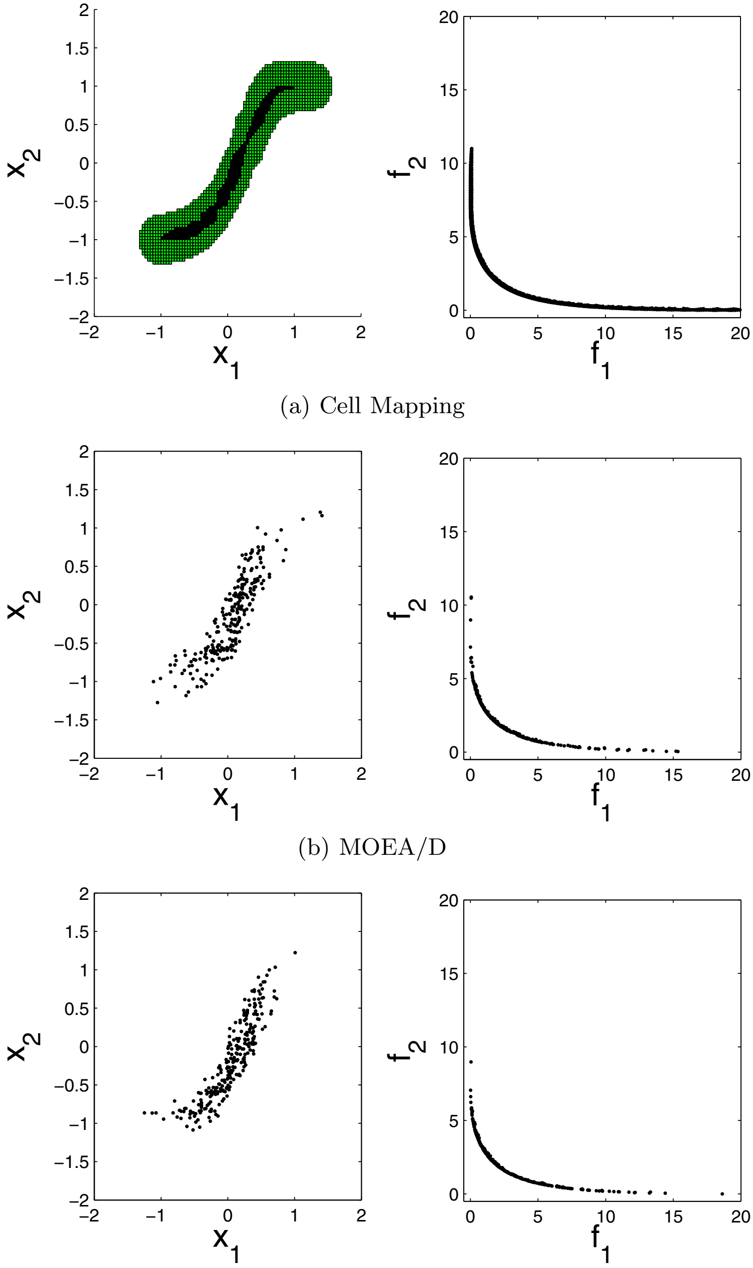
$$\delta_t = \begin{cases} 0 & \text{for } t_1 = 0 \text{ and } t_2 = 0 \\ 0.1 & \text{else} \end{cases}.$$

Using $a = 0.5$, $b = 5$, $c = 5$ the Pareto set of MOP (8) is connected and further there exist eight other connected components that are locally optimal. For $\epsilon > 0.1$, the set $P_{Q,\epsilon}$ consists of nine sets that contain these components. Figure 4 shows some numerical results. SCM computes a covering of the entire set of interest, while the evolutionary strategies do not always detect all components which is e.g. reflected by the averaged Δ_p values in Tables 1 and 2.

As a hypothetical decision making problem assume the DM is interested in the performance $Z = [0.2132, 0.2932]$ (measured in objective space) and is willing to accept a deterioration of $\epsilon = [0.1, 0.1]$. Then, e.g. the representatives of the cells those images are within the target regions can be presented to the DM leading here to the following 22 candidate solutions:

$$\begin{array}{llll} (-6.04,-5.00), & (-0.04,-5.00), & (5.96,-5.00), & (-6.04,-0.04), \\ (-0.12,-0.04), & (-0.04,-0.28), & (-0.04,-0.20), & (-0.04,-0.12), \\ (-0.04,-0.04), & (-0.12,0.04), & (-0.04,0.04), & (-0.04,0.12), \\ (-0.04,0.20), & (-0.04,0.28), & (0.04,-0.12), & (0.04,-0.04), \\ (0.04,0.04), & (0.04,0.12), & (5.96,-0.04), & (-6.04,5.00), \\ (-0.04,5.00), & (5.96,5.00). & & \end{array}$$

The solutions are well-spread and come in this case from all nine components of $P_{Q,\epsilon}$. Since these components are located in different regions of the parameter space, the DM is hence given a large variety for the realization of his/her project.

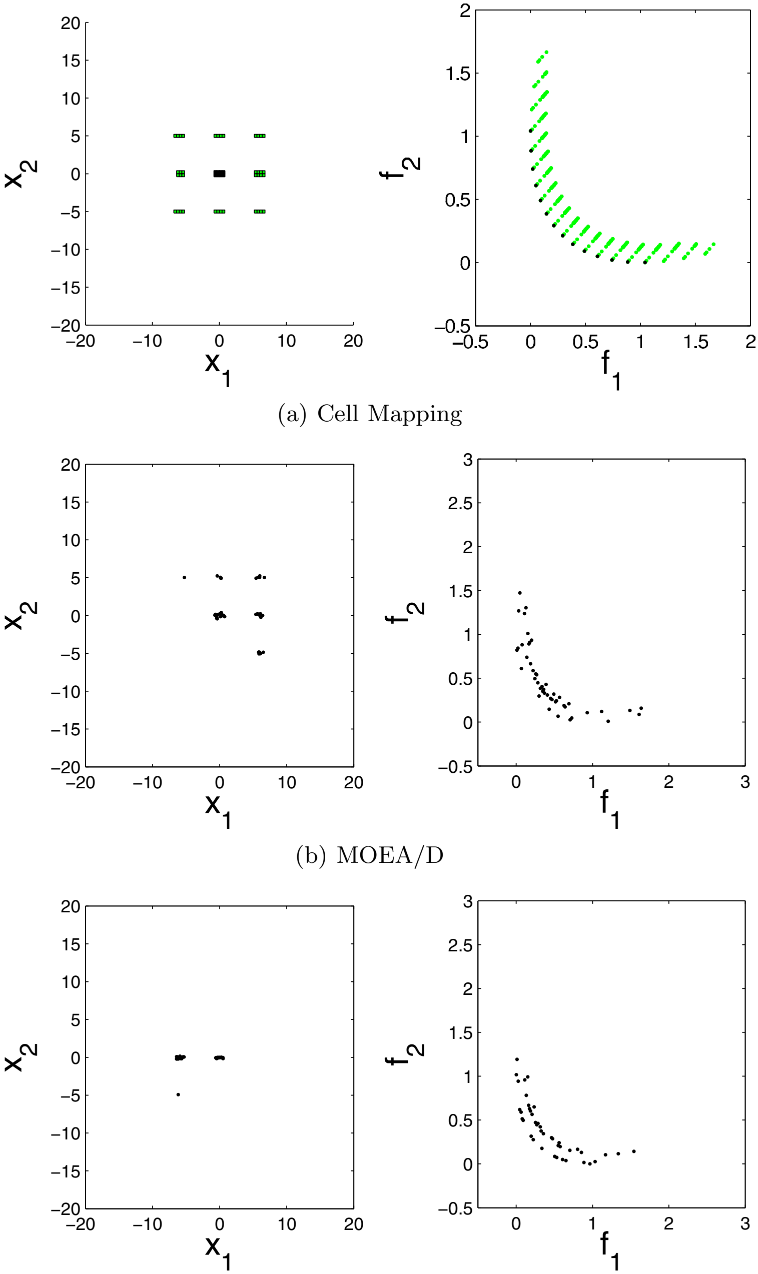


(a) Cell Mapping

(b) MOEA/D

(c) NSGA-II

Fig. 3. Numerical results for MOP (7). Black cells indicate Pareto optimal candidates and yellow cells regions in $P_{Q,\epsilon}$ that are not optimal.

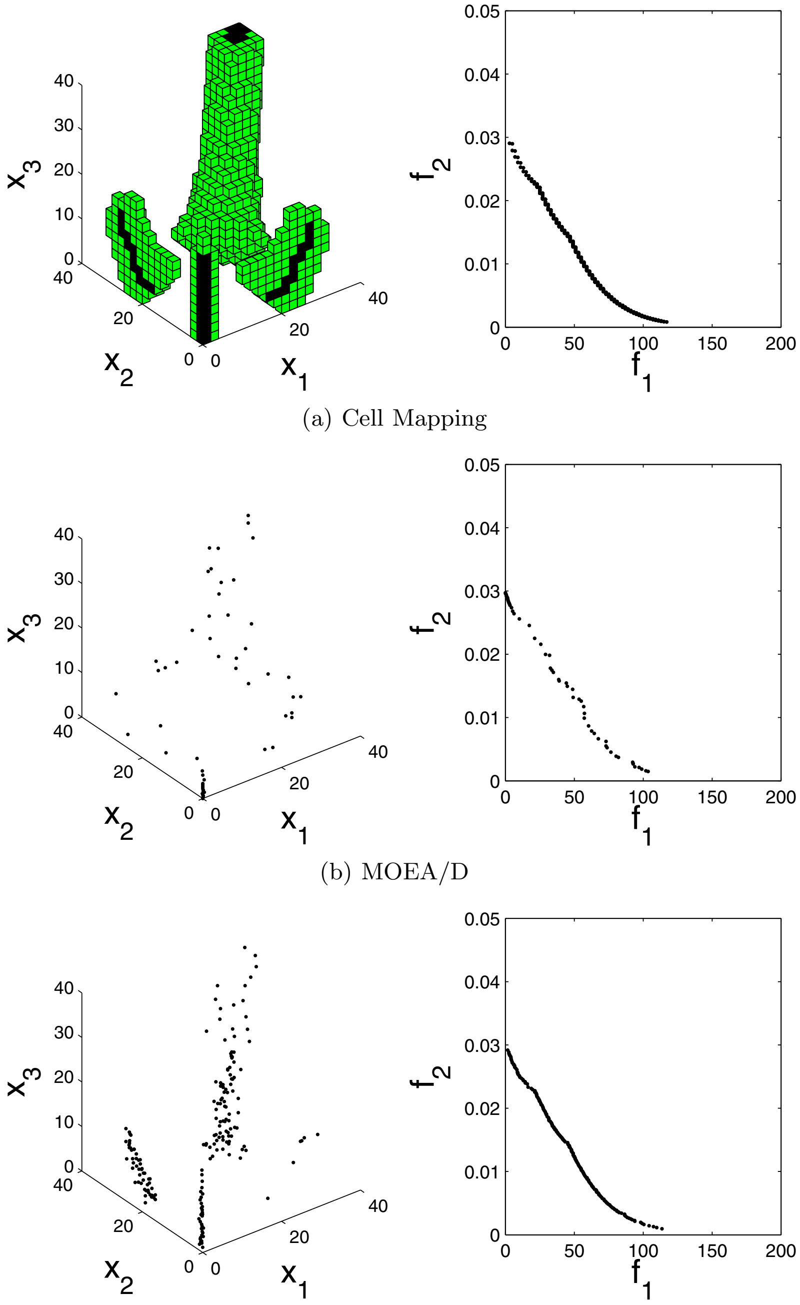


(a) Cell Mapping

(b) MOEA/D

(c) NSGA-II

Fig. 4. Numerical results for MOP (8). Black cells indicate Pareto optimal candidates and yellow cells regions in $P_{Q,\epsilon}$ that are not optimal.



(a) Cell Mapping

(b) MOEA/D

(c) NSGA-II

Fig. 5. Numerical results for MOP (9). Black cells indicate Pareto optimal candidates and yellow cells regions in $P_{Q,\epsilon}$ that are not optimal.

4.3 Problem 3

Finally, we consider the production model proposed in [29]:

$$\begin{gathered} f_1, f_2 : \mathbb{R}^3 \to \mathbb{R} \\ f_1(x) = \sum_{j=1}^{3} x_j, \quad f_2(x) = 1 - \prod_{j=1}^{3}(1 - w_j(x_j)), \end{gathered} \tag{9}$$

where

$$w_j(z) = \begin{cases} 0.01 \cdot \exp(-(\frac{z}{20})^{2.5}) & \text{for} \quad j = 1,2 \\ 0.01 \cdot \exp(-\frac{z}{15}) & \text{for} \quad j = 3 \end{cases}$$

Objective f_1 is related to the cost of a given product and f_2 to its failure rate. For the domain $Q = [0, 40]^3$ the Pareto set consists of four connected components.

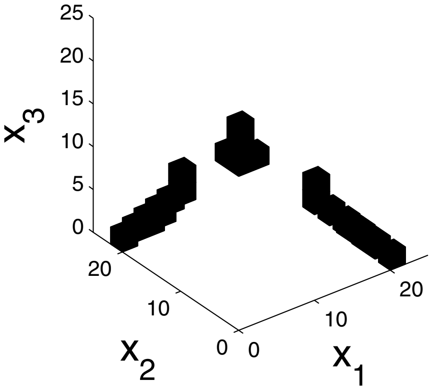


Fig. 6. Potential solutions for the hypothetical decision making problem related to MOP (8)

Table 1. Δ_p values for the distances of the candidate solution set to $P_{Q,\epsilon}$, the best solutions in boldface. For MOP (7) (Convex) a budget of 10,000 function evaluations was used, the grid for SCM was set to $N = [40, 40]$. For MOP (8) (Rudolph) and MOP (9) (Schäffler), a budget of 60,000 function evaluations was used, the grid for SCM in Rudolph was $N = [100, 100]$ and the grid for SCM in Schäffler was $N = [20, 20, 20]$. The results of the evolutionary computations are averaged over 20 independent runs.

	MOEA/D	NSGA-II	SCM
Convex	0.5141	0.4628	**0.0849**
Rudolph	5.0507	7.4737	**0.0632**
Schäffler	10.8365	10.9384	**0.8660**

Figure 5 shows a numerical result of the SCM and the evolutionary strategies. Also in this case, SCM obtains the best result in particular measured in parameter space (compare also to Tables 1 and 2). Figure 6 shows the resulting boxes of another hypothetical decision making problem where we have chosen $Z = [23, 0.02231]$ and $\epsilon = [2, 0.0004]$. Also here, the DM is offered an entire range of solutions with different parameter values.

Table 2. Δ_p values for the distances of the images of the candidate sets to $F(P_{Q,\epsilon})$, see Table 1 for details

	MOEA/D	NSGA-II	SCM
Convex	7.8902	8.0027	**2.4250**
Rudolph	0.4276	0.6317	**0.0524**
Schäffler	5.8152	2.6852	**1.5000**

5 Conclusion and Future Work

In this paper we have addressed the problem of computing the set of approximate solutions of a given multi-objective optimization problem. This set is of interest for the decision maker since it might enhance the set of options for him/her when compared to the set of optimal solutions, the Pareto set. To compute the set of approximate solutions we have adapted cell mapping techniques that were originally designed for the global analysis of dynamical systems. Since methods of that kind divide the search space into n-dimensional cells, where n is the dimension of the decision space of the MOP, they are well-suited for the problem at hand since they allow for an efficient approximation of the set of interest. We have tested the novel method on three academic functions and have compared it against two evolutionary methods. The results indicate that the cell mapping technique is able to reliably compute the set of approximate solutions, and is faster than the evolutionary approaches. The method, however, is restricted to small dimensions of the decision space.

For future work, there are several points to be addressed. First of all, it would be desirable to extend the applicability of the method to higher dimensional problems. For this, it seems promising to couple the cell mapping techniques with related set oriented methods such as subdivision techniques ([11,13]) or recovering algorithms ([13,33]). Next, the integration of constraint handling techniques has to be addresses which has been left out in this study. Finally, we plan to apply the new method to real-wold engineering problems.

Acknowledgement. The authors acknowledge support from the UC MEXUS-CONACyT project "*Cell-to-cell Mapping for Global Multi-objective Optimization.*" The first author acknowledges support from CONACyT through a scholarship to pursue graduate studies at the Computer Science Department of CINVESTAV-IPN.

References

1. Lara, A.: Using Gradient Based Information to build Hybrid Multi-objective Evolutionary Algorithms. PhD thesis (2012)
2. Blesken, M., Chebil, A., Rückert, U., Esquivel, X., Schütze, O.: Integrated circuit optimization by means of evolutionary multi-objective optimization. In: Genetic and Evolutionary Computation Conference (GECCO 2011), pp. 807–812 (2011)
3. Blesken, M., Rückert, U., Steenken, D., Witting, K., Dellnitz, M.: Multiobjective optimization for transistor sizing of CMOS logic standard cells using set-oriented numerical techniques. In: 27th Norchip Conference (2009)
4. Bursal, F.H., Hsu, C.S.: Application of a cell-mapping method to optimal control problems. International Journal of Control 49(5), 1505–1522 (1989)
5. Coello Coello, C.A., Lamont, G., Van Veldhuizen, D.: Evolutionary Algorithms for Solving Multi-Objective Problems, 2nd edn. Springer (2007)
6. Coello Coello, C.A., Cruz Cortés, N.: Solving multiobjective optimization problems using an artificial immune system. Genetic Programming and Evolvable Machines 6(2), 163–190 (2005)
7. Crespo, L.G., Sun, J.Q.: Stochastic optimal control of nonlinear dynamic systems via bellman's principle and cell mapping. Automatica 39(12), 2109–2114 (2003)
8. Das, I., Dennis, J.: Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems. SIAM Journal of Optimization 8, 631–657 (1998)
9. Deb, K.: Multi-Objective Optimization Using Evolutionary Algorithms. Wiley (2001)
10. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA–II. IEEE Transactions on Evolutionary Computation 6(2), 182–197 (2002)
11. Dellnitz, M., Hohmann, A.: A subdivision algorithm for the computation of unstable manifolds and global attractors. Numerische Mathematik 75, 293–317 (1997)
12. Dellnitz, M., Ober-Blöbaum, S., Post, M., Schütze, O., Thiere, B.: A multiobjective approach to the design of low thrust space trajectories using optimal control. Celestial Mechanics and Dynamical Astronomy 105, 33–59 (2009)
13. Dellnitz, M., Schütze, O., Hestermeyer, T.: Covering Pareto sets by multilevel subdivision techniques. Journal of Optimization Theory and Applications 124, 113–155 (2005)
14. Eichfelder, G.: Adaptive Scalarization Methods in Multiobjective Optimization. Springer, Heidelberg (2008) ISBN 978-3-540-79157-7
15. Fliege, J.: Gap-free computation of Pareto-points by quadratic scalarizations. Mathematical Methods of Operations Research 59, 69–89 (2004)
16. Heinonen, J.: Lectures on Analysis on Metric Spaces. Springer, New York (2001)
17. Hillermeier, C.: Nonlinear Multiobjective Optimization - A Generalized Homotopy Approach. Birkhäuser (2001)
18. Hsu, C.S.: A theory of cell-to-cell mapping dynamical systems. Journal of Applied Mechanics 47, 931–939 (1980)
19. Hsu, C.S.: A discrete method of optimal control based upon the cell state space concept. Journal of Optimization Theory and Applications 46(4), 547–569 (1985)
20. Hsu, C.S.: Cell-to-cell mapping: a method of global analysis for nonlinear systems. In: Applied Mathematical Sciences. Springer (1987)
21. Fliege, J., Svaiter, B.F.: Steepest Descent Methods for Multicriteria Optimization. Mathematical Methods of Operations Research 51(3), 479–494 (2000)

22. Jahn, J.: Multiobjective search algorithm with subdivision technique. Computational Optimization and Applications 35(2), 161–175 (2006)
23. Klamroth, K., Tind, J., Wiecek, M.: Unbiased approximation in multicriteria optimization. Mathematical Methods of Operations Research 56, 413–437 (2002)
24. Loridan, P.: ϵ-solutions in vector minimization problems. Journal of Optimization, Theory and Application 42, 265–276 (1984)
25. Schütze, O., Lara, A., Coello Coello, C.A.: The Directed Search Method for Unconstrained Multi-Objective Optimization Problems. Technical Report COA-R1, CINVESTAV-IPN (2010)
26. Schütze, O., Laumanns, M., Tantar, E., Coello Coello, C.A., Talbi, E.-G.: Computing Gap Free Pareto Front Approximations with Stochastic Search Algorithms. Evolutionary Computation 18(1)
27. Pareto, V.: Manual of Political Economy. The MacMillan Press (1971); original edition in French (1927)
28. Rudolph, G., Naujoks, B., Preuß, M.: Capabilities of emoa to detect and preserve equivalent pareto subsets. In: Obayashi, S., Deb, K., Poloni, C., Hiroyasu, T., Murata, T. (eds.) EMO 2007. LNCS, vol. 4403, pp. 36–50. Springer, Heidelberg (2007)
29. Schäffler, S., Schultz, R., Weinzierl, K.: A stochastic method for the solution of unconstrained vector opimization problems. Journal of Optimization, Theory and Application 114(1), 209–222 (2002)
30. Schütze, O.: Set Oriented Methods for Global Optimization. PhD thesis, University of Paderborn (2004), http://ubdata.uni-paderborn.de/ediss/17/2004/schuetze/
31. Schütze, O., Coello Coello, C.A., Talbi, E.-G.: Approximating the ϵ-efficient set of an MOP with stochastic search algorithms. In: Gelbukh, A., Kuri Morales, Á.F. (eds.) MICAI 2007. LNCS (LNAI), vol. 4827, pp. 128–138. Springer, Heidelberg (2007)
32. Schütze, O., Coello Coello, C.A., Tantar, E., Talbi, E.-G.: Computing finite size representations of the set of approximate solutions of an MOP with stochastic search algorithms. In: GECCO 2008: Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation, pp. 713–720. ACM, New York (2008)
33. Schütze, O., Dell'Aere, A., Dellnitz, M.: On continuation methods for the numerical treatment of multi-objective optimization problems. In: Branke, J., Deb, K., Miettinen, K., Steuer, R.E. (eds.) Practical Approaches to Multi-Objective Optimization. Dagstuhl Seminar Proceedings, vol. 04461. Internationales Begegnungs- und Forschungszentrum (IBFI), Schloss Dagstuhl, Germany (2005), http://drops.dagstuhl.de/opus/volltexte/2005/349
34. Schütze, O., Esquivel, X., Lara, A., Coello Coello, C.A.: Using the averaged Hausdorff distance as a performance measure in evolutionary multi-objective optimization. IEEE Transactions on Evolutionary Computation 16(4), 504–522 (2012)
35. Schütze, O., Vasile, M., Coello Coello, C.A.: Computing the set of epsilon-efficient solutions in multiobjective space mission design. Journal of Aerospace Computing, Information, and Communication 8, 53–70 (2011)
36. Schütze, O., Vasile, M., Junge, O., Dellnitz, M., Izzo, D.: Designing optimal low thrust gravity assist trajectories using space pruning and a multi-objective approach. Engineering Optimization 41, 155–181 (2009)
37. Schütze, O., Vasile, M., Junge, O., Dellnitz, M., Izzo, D.: Designing optimal low thrust gravity assist trajectories using space pruning and a multi-objective approach. Engineering Optimization 41(2), 155–181 (2009)

38. Van Veldhuizen, D.A.: Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations. PhD thesis, Department of Electrical and Computer Engineering. Graduate School of Engineering. Air Force Institute of Technology, Wright-Patterson AFB, Ohio (May 1999)
39. Zhang, Q., Li, H.: MOEA/D: A multi-objective evolutionary algorithm based on decomposition. IEEE Transactions on Evolutionary Computation 11(6), 712–731 (2007)