MILP models for objective reduction in multi-objective optimization: Error measurement considerations and Non-Redundancy ratio

Daniel Vázquez , Rubén Ruiz-Femenia , Laureano Jiménez , José A. Caballero

PII: S0098-1354(18)30384-3
DOI: 10.1016/j.compchemeng.2018.04.031
Reference: CACE 6099

To appear in: *Computers and Chemical Engineering*

Received date: 15 January 2018
Revised date: 20 April 2018
Accepted date: 26 April 2018

Highlights

- Measurement of the error in which a system incurs when removing one objective using a norm 1.
- A new model introducing a new concept considering a modification on the solutions instead of the objectives.
- Systematic characterization of a system in regards of its objectives and Pareto solutions.
- Combination with other techniques, such as the Principal Component Analysis.

MILP models for objective reduction in multi-objective optimization: Error measurement considerations and Non-Redundancy ratio.

Daniel Vázquez [a,b], Rubén Ruiz-Femenia[a], Laureano Jiménez[b], José A. Caballero[a]

[a] *Institute of Chemical Processes Engineering, University of Alicante, PO 99, E-3080 Alicante, Spain*
[b] *Department d'Enginyeria Quimica, Universitat Rovira i Virgili, Av. Països Catalans 26, 43007 Tarragona, Spain*

**caballer@ua.es*

Abstract

A common approach in multi-objective optimization (MOO) consists of removing redundant objectives or reducing the set of objectives minimizing some metrics related with the loss of the dominance structure. In this paper, we comment some weakness related to the usual minimization of the maximum error (infinity norm or δ-error) and the convenience of using a norm 1 instead. Besides, a new model accounting for the minimum number of Pareto solutions that are lost when reducing objectives is provided, which helps to further describe the effects of the objective reduction in the system. A comparison of the performance of these algorithms and its usefulness in objective reduction against principal component analysis + Deb & Saxena's algorithm (Deb & Saxena Kumar, 2005) is provided, and the ability of combining it with a principal component analysis in order to reduce the dimensionality of a system is also studied and commented.

Keywords: MOO objective reduction, Non-Redundancy ratio, PCA, δ-error, Deb & Saxena algorithm.

1. Introduction

Nowadays, one of the main objectives of the process system engineering community is to develop methods to assess a problem from a holistic point of view. As such, multi-criteria analysis as opposed to the traditional economic focused analysis is rising among researchers and industries. This does not mean that the economic point of view has lost significance, but that other subjects, especially environmental, social and safety assessment of a chemical process, have been taking part of the spotlight over the last years. There are multiple alternatives to deal with multi-criteria analysis of a problem, such as Analytical Hierarchical Process (AHP) (Saaty, 1990), Analytic Network Process (ANP) (Saaty, 2001), Case Base Reasoning (Aamodt & Plaza, 1994), Data Envelopment Analysis (DEA) (Banker et al., 1984) and fuzzy set theory (Zadeh, 1965) among others.

Multi-objective optimization (MOO) has proven to be an excellent method to simultaneously take into account different criteria to assess a process when seeking for an optimum design or result of a problem. Aspects such as the economic, environmental or social ones must be considered when providing an optimal design for a plant from a holistic point of view, or even for smaller problems such as a piece of equipment. A huge advantage of these methods is that the solution resultant from their use is a set of Pareto optimal alternatives. If a Pareto optimal alternative exists, it is ensured that no other feasible alternative will be better in any objective without worsening one or more of the remaining objectives. All these Pareto optimal points are equally “good” in terms of optimality, but techniques to differentiate them do exist, such as the efficiency of order k (Das, 1999), which is able to classify Pareto solutions in different orders, thus helping the decision maker to choose among a smaller pool of equally good solutions.

An important problem when using MOO techniques is the computational burden. A well-known method, such as the ε-constraint method (Haimes et al., 1971) increases its number of calculations in an order of magnitude per objective taken into account. Hence, problems with more than 3 or 4 objectives represent a serious difficulty in calculations. A simple, direct alternative to problems with high dimensionality of objectives is the use of aggregated objectives. This is common practice in the environmental area, with aggregated metrics such as the Econindicator-99 (PRé-Consultants, 2000), ReCiPe (Goedkoop et al., 2009) and IMPACT 2002+ (Jolliet et al., 2003). However, this approach leads to many problems. On one hand, the aggregation of impacts is “subjective”, even though most of these aggregations are based on weights determined by a panel of experts. The relative importance of each objective is in most cases at least arguable, since they can be case dependent and the criteria of the final user (manager or designer) can be different. On the other hand, the dominance structure of the problem may be altered (Guillén-Gosálbez, 2011), and as such, when there exists an aggregation of objectives, there may be differences in the results when considering that aggregation versus the initial objectives (Carreras et al., 2016). Due to these problems, a strategy of reducing the dimensionality of a problem without impacting the dominance structure is preferred and necessary when working with MOO problems.

In a previous work (Vázquez et al., 2018), we proposed three models which allowed the user to minimize the number of objectives by maintaining the dominance structure of the system to a certain degree, given some inputs. It is based upon the concept of δ-error, introduced by Brockhoff & Zitzler (2006a, 2006b, 2006c, 2009) and on the work performed by Guillén-Gosálbez (2011), increasing its efficiency and its modifiability in order to obtain different models which allow the user to perform different studies. The first model is able to remove redundant objectives without incurring any error and retaining the dominance structure intact. The second model allows removing objectives that are redundant with a certain tolerance added to the δ-error. This problem is also denominated the δ-Minimum Objective Subset (δ-MOSS) problem. The third model allows the user to find a subset of at most *k* objectives with the smallest possible δ-error. This is also denominated the Minimum Objective Subset of Size k with Minimum Error (k-EMOSS) problem. While the full-fledged explanation of these methods is given in the previous work, an overview of the basic theoretical foundation is revised below, where the main concepts are revisited. The gist of the models is to allow the user to discriminate among different objectives with a proper mathematical measurement of the error incurred when an objective is removed. The necessary data for these models are already Pareto optimal points in the whole space of initial objectives, which may seem counter-intuitive at first. A common drawback of the objective reduction methods is that an exploration of the Pareto frontier of the problem in the whole space of initial objectives is required. However, this exploration must only be performed up until a significant number of Pareto points able to characterize the Pareto frontier is obtained. Thus, with selected data from the initial space of objectives, we can rank these objectives in "importance" (i.e., the error we incur when we remove them), and we can decide to remove the less important ones for the dominance structure. By doing this, the user can now explore the reduced space of objectives more thoroughly and perform a multi-objective optimization in a much more maneuverable way.

In our previous work, we mentioned that while the second and third model provided satisfactory solutions, it should be taken into account that the δ-error was computed with an infinity norm. As such, a result of the model could present a small δ-error, but this error would be present in multiple Pareto solutions. Another possible outcome is the possibility of obtaining a big δ-error, caused solely by a couple of Pareto solutions. This behavior is good enough for the majority of the problems, but a model that allows us to obtain the minimum number of Pareto solutions that have to be violated in order to maintain completely the dominance structure (i.e., δ-Error = 0) can be useful, since it provides a different approximation to define the structure of a problem. Therefore, we introduce a new metric denominated **non-redundancy ratio,** whose model is developed below in the paper. This concept is not intended to substitute the δ-error in regards of reducing objectives, but to complement it. Since the previous models were written to take into account only an infinity norm, we also introduce the necessary modifications to the models in order to use a 1-norm approach instead, exemplifying how this consideration would change the results and the related benefits. The model for 1-norm can be easily extended to every p-norm.

Another well-known method to reduce the dimensionality of a problem is the Principal Component Analysis (PCA) (Pearson, 1901). It has been used before with this end by other researchers, see for example (Pozo et al., 2012), and its effectivity when combined with the algorithm of Deb and Saxena Kumar (2005) to reduce objectives in the original space of objectives has been thoroughly proved. In this paper, we compare the PCA method with our models and provide some guidelines to use them in conjunction to obtain the best possible results. In addition, we compare the performance obtained by the combination of PCA and Deb's algorithm with our model results.

The paper presents the following structure. The next section is an overview of the main concepts, as well as a recapitulation of mathematical programming formulations of the redundant objectives removal, δ-MOSS and k-EMOSS models. The following section introduces the concept of non-redundancy ratio, which allows us to further classify subspaces of objectives. A brief explanation of the PCA and the Deb & Saxena algorithm follows, before the case studies utilized to test our algorithms. The results from these cases are then presented, followed by a conclusion section that closes the work.

2. Theoretical foundations

A brief overview of the theoretical foundations is presented. A comprehensive review of the theoretical foundations is out of the scope of this paper, but the interested reader is referred to our previous work (Vázquez et al., 2018) and references therein.

Consider a decision space X defined over a set F of k objective functions: $f_i : X \to \mathbb{R}$, $i \in F$ that must be minimized. A solution $\mathrm{x} \in X$ is said to **weakly dominate** another solution $\mathrm{y} \in X$ if and only if x is not worse than y in all the objectives.

In this paper, in order to avoid the repetition, we will refer to the $f_i(\mathrm{x}) := \left\{i \in F, \mathrm{x} \in X\right\}$ as $x_{s,i}$, where $s \in X$ and $i \in F$. This can be done because the paper centers the models in the value of the points in those objective functions, instead of the values of the proper points. As such, it is said that a solution s weakly dominates another solution s' in a **minimization problem** if $x_{s,i} \le x_{s',i} \ \ \forall i \in F$.

The dominance structure of a problem represents the weakly dominances between a set of objectives for a set of solutions, since it is rare that we can evaluate the whole space of solutions X. As such, we call $S \subseteq X$ to the subspace of individual solutions s. Since we cannot either evaluate all the objectives in the world, we call $OBJ \subseteq F$ to our initial set of objectives. In Figure 1(a) we have the subsets of objectives $OBJ := \left\{f_1, f_2, f_3\right\}$ and the subset of solutions $S := \left\{A, B\right\}$. In this parallel plot, where each point refers to $x_{s,i}$, it can be seen that $x_{B,i} \le x_{A,i} \quad \forall i \in \left\{f_1, f_2\right\}$ and as such B weakly dominates A in the subset of $OBJ' := \left\{f_1, f_2\right\}$. As shown in Figure 1 (b), if we consider this subset of objectives, we would incur in a δ-error of 0.4. If we consider the subset $OBJ'' := \left\{f_1, f_3\right\}$, as shown in Figure 1 (c) or even the subset $OBJ''' := \left\{f_2, f_3\right\}$, the δ-error would be zero. This is because by choosing these subsets of objectives, not a single solution becomes weakly dominated by the others.

< Fig.1 >

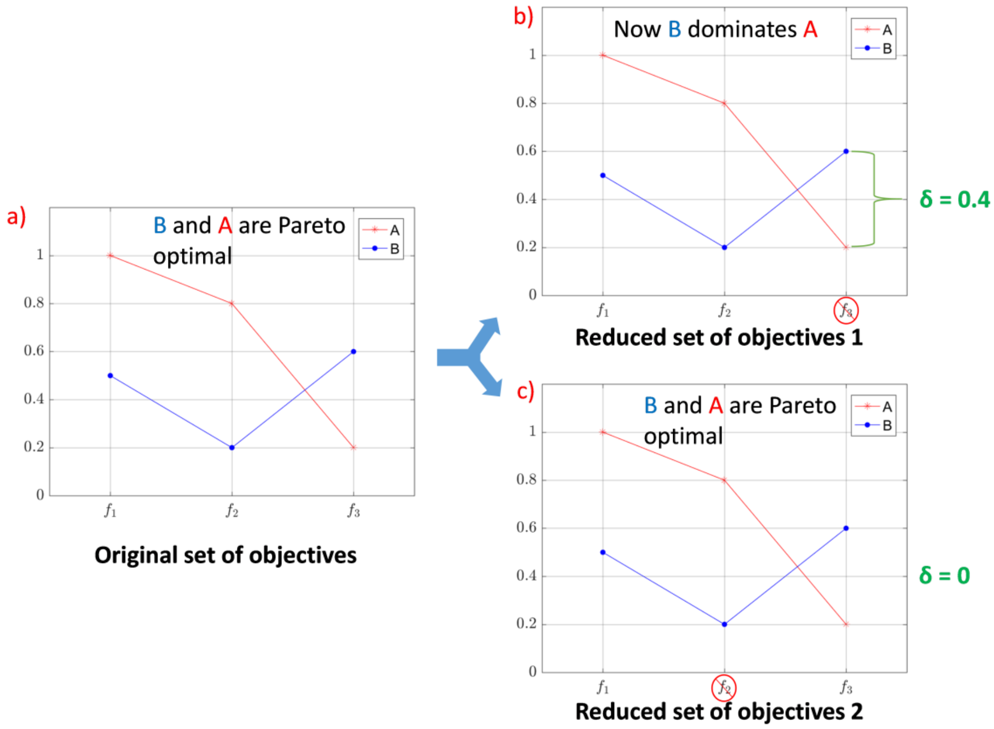

Figure 1: Parallel plot example. a) 3 initial objectives b) Objectives 1 and 2 maintained c) Objectives 1 and 3 maintained

The δ-error can be then defined as the value of $|x_{s',i'} - x_{s,i'}| \quad \forall i' \in OBJ \setminus OBJ'$ if $x_{s,i} \leq x_{s',i} \ \forall i \in OBJ'$. In simple terms, it is the value of the difference between the solutions in the objective that is being removed, when by removing this objective we are making a solution weakly dominated by another solution.

An objective that can be reduced without affecting the dominance structure of a problem is called a **redundant objective**. In the example shown in Figure 1, both f_1 and f_2 are redundant (not the subset $\{f_1, f_2\}$ though). If you remove one of them, the δ-error of the two remaining is zero. The first model, which takes the form of a set covering problem, allows us to choose the minimum number of **non-redundant objectives**. It is shown in Model 1.

$$\begin{aligned} \min & \sum_{i \in OBJ} y_i \\ s.t. & \sum_{i \in OBJ} C_{1,s,s',i} \; y_i \geq 1 \quad \forall (s,s') \in P_{s,s'} \\ & \sum_{i \in OBJ} C_{2,s,s',i} \; y_i \geq 1 \quad \forall (s,s') \in P_{s,s'} \\ & y_i \in \{0,1\} \end{aligned} \qquad \text{(Model 1)}$$

In this model, y_i refers to a binary variable that takes the value 1 when the objective i is chosen. The subset $P_{s,s'}$ is formed by the pairs of solutions $s,s' : \{s' > s\}$. We must point out that this is possible due to the fact that S is an ordered set. The parameters $C_{1,s,s',i}$ and $C_{2,s,s',i}$ are calculated as shown in Eq.(1).

$$\begin{aligned} C_{1s,s',i} &= 1 \qquad \forall (P_{s,s'}, i \in OBJ) \mid (x_{s',i} - x_{s,i}) > 0 \\ C_{1s,s',i} &= 0 \qquad \forall (P_{s,s'}, i \in OBJ) \mid (x_{s',i} - x_{s,i}) \leq 0 \\ \\ C_{2s,s',i} &= 1 \qquad \forall (P_{s,s'}, i \in OBJ) \mid (x_{s',i} - x_{s,i}) \leq 0 \\ C_{2s,s',i} &= 0 \qquad \forall (P_{s,s'}, i \in OBJ) \mid (x_{s',i} - x_{s,i}) > 0 \end{aligned} \qquad (1)$$

When we find ourselves with objectives that have very different orders, such as an economic one and any environmental objective, the data $x_{s,i}$ must be scaled. If the limits of each objective are not known, it is recommended to use the maximum and minimum of each objective in the studied data to scale the values of the others. Bear in mind that by scaling with the maximum and minimum values found, the scaling will be directly affected by how effectively the search domain has been explored. This is not important for this algorithm, since assuming δ-error = 0 will avoid any scaling problem, but when we want to allow a certain δ-error, it will greatly depend on the scaling of the system.

Our second model allows the user to obtain a subset of objectives which are δ-minimum with relation to the original set of objectives. By δ-minimum we mean that the dominance structure was broken at most a previously defined δ value δ_{MAX} in the subset of objectives. This is known as the δ-MOSS problem. It is shown in Model 2.

$$
\begin{aligned}
&\min z = \sum_{i \in OBJ} y_i \\
&s.t. \\
&\left.\begin{aligned}
&\sum_{i \in OBJ} C_{1,s,s',i} y_i + v_{1s,s'} \geq 1 \\
&\sum_{i \in OBJ} C_{2s,s',i} y_i + v_{2s,s'} \geq 1
\end{aligned}\right\} \forall (s,s') \in P_{s,s'} \\
&\quad \delta \leq \delta_{MAX} \\
&\left.\begin{aligned}
&\delta \geq Pen_{1s,s',i}\, w_{1s,s',i} \\
&\delta \geq Pen_{2s,s',i}\, w_{2s,s',i} \\
&y_i + (1 - v_{1s,s'}) + w_{1s,s',i} \geq 1 \\
&y_i + (1 - v_{2s,s'}) + w_{2s,s',i} \geq 1 \\
&(1 - y_i) + (1 - w_{1s,s',i}) \geq 1 \\
&(1 - y_i) + (1 - w_{2s,s',i}) \geq 1 \\
&v_{1s,s'} + 1 - w_{1s,s',i} \geq 1 \\
&v_{2s,s'} + 1 - w_{2s,s',i} \geq 1
\end{aligned}\right\} \forall (s,s') \in P_{s,s'},\ \forall i \in OBJ \\
&y_i, v_{1,s,s'}, v_{2,s,s'} \in \{0,1\}
\end{aligned}
\qquad \text{(Model 2)}
$$

Briefly explained, since it is detailed in the previous work, we introduce binary variables $v_{1,s,s'}, v_{2,s,s'}$ which allow to break the set covering restrictions. We introduce pseudo binary variables $w_{1,s,s',i}, w_{2,s,s',i}$ which allow us to relate the binary variables y_i and v_1, v_2, and the δ-error is calculated as the infinity norm with the help of previously calculated parameters $Pen1_{s,s',i}, Pen2_{s,s',i}$, which are calculated by using Eq.(2).

$$
\begin{aligned}
&Pen_{1,s,s',i} = x_{s',i} - x_{s,i} && \forall (P_{s,s'}, i) \mid (x_{s',i} \geq x_{s,i}) \\
&Pen_{1,s,s',i} = 0 && \forall (P_{s,s'}, i) \mid (x_{s',i} < x_{s,i}) \\
\\
&Pen_{2,s,s',i} = x_{s,i} - x_{s',i} && \forall (P_{s,s'}, i) \mid (x_{s',i} < x_{s,i}) \\
&Pen_{2,s,s',i} = 0 && \forall (P_{s,s'}, i) \mid (x_{s',i} \geq x_{s,i})
\end{aligned}
\qquad (2)
$$

The third model allows the user to fix the cardinality of the subset of objectives desired and obtain a δ-minimum subset which also has minimum δ-value. This is known as the k-EMOSS problem. It is shown in Model 3.

$$
\begin{aligned}
&\min z = \delta \\
&s.t. \\
&\quad \sum_{i \in OBJ} y_i = Nobj \\
&\left.\begin{aligned}
&\sum_{i \in OBJ} C_{1,s,s',i} \cdot y_i + v_{1s,s'} \geq 1 \\
&\sum_{i \in OBJ} C_{2s,s',i} \cdot y_i + v_{2s,s'} \geq 1
\end{aligned}\right\} \forall (s,s') \in P_{s,s'} \\
&\left.\begin{aligned}
&\delta \geq Pen_{1s,s',i} \cdot w_{1s,s',i} \\
&\delta \geq Pen_{2s,s',i} \cdot w_{2s,s',i} \\
&y_i + (1 - v_{1s,s'}) + w_{1s,s',i} \geq 1 \\
&y_i + (1 - v_{2s,s'}) + w_{2s,s',i} \geq 1 \\
&(1 - y_i) + (1 - w_{1s,s',i}) \geq 1 \\
&(1 - y_i) + (1 - w_{2s,s',i}) \geq 1 \\
&v_{1s,s'} + 1 - w_{1s,s',i} \geq 1 \\
&v_{2s,s'} + 1 - w_{2s,s',i} \geq 1
\end{aligned}\right\} \forall (s,s') \in P_{s,s'},\ \forall i \in OBJ \\
&\quad y_i, v_{1,s,s'}, v_{2,s,s'} \in \{0,1\}
\end{aligned}
\qquad \text{(Model 3)}
$$

We set the cardinality with the parameter $Nobj$. Both Model 2 and Model 3 work with the assumption of an infinity norm. This being, the δ-error of the system is the highest δ-error that exists in it.

3. Modification to the measurement of δ-error and NR-ratio

Up until now, the models developed were using only as measure the maximum δ-error of the system as measure of the alteration of the dominance structure. As an alternative, the use of a Norm 1 to define the error of a system is proposed. Another concept, the non-redundancy ratio, is introduced as well as a complementary measure of the error in the dominance structure.

3.1 δ-error of Norm 1

An infinity norm to estimate the error induced in the dominance structure by removing some objectives, focus only on the maximum error, but it has at least an important weakness:

It is possible to select a subset of objectives with a large number of 'breaks' in the dominance structure induced by a single solution. In that case it is possible to argue that a set of objectives that maintain the dominance structure in most of the Pareto surface, even with a larger error in a point, is preferable to a set of objectives that do not maintain the dominance structure in most of the Pareto surface. A norm 1 is a good option in this case because it takes into account all the errors induced by removed objectives and at the same time larger errors have larger impacts in the objective function.

If we take the third model (k-EMOSS) (or the second, -δ-MOSS-), the infinity norm is given by equations that define the δ-error, which are shown in Eq.(3)

$$\left.\begin{array}{l} \delta \geq Pen_{2s,s',i} \cdot w_{2s,s',i} \\ \delta \geq Pen_{2s,s',i} \cdot w_{2s,s',i} \end{array}\right\} \forall (s,s') \in P_{s,s'}, \forall i \in OBJ \tag{3}$$

If we introduce a new variable, $\bar{\delta}_i$, defined as the maximum δ-error for an objective i, we can simply rewrite Eq.(3) as Eq.(4).

$$\left.\begin{array}{l} \bar{\delta}_i \geq Pen_{2s,s',i} \cdot w_{2s,s',i} \\ \bar{\delta}_i \geq Pen_{2s,s',i} \cdot w_{2s,s',i} \end{array}\right\} \forall (s,s') \in P_{s,s'}, \forall i \in OBJ \tag{4}$$

Now we only need to change the objective function accordingly and we obtain our third Model expressed as norm 1, shown in Model 3.1. In order to normalize the error and avoid having numbers higher than 1, we divide its sum by the number of maintained objectives. This step is optional if only for representing the error. It's better to perform the optimization without it.

$$\begin{array}{l} \min z = \sum\limits_{i \in OBJ} \bar{\delta}_i \\ s.t. \\ \quad \sum\limits_{i \in OBJ} y_i = Nobj \\ \quad \left.\begin{array}{l} \sum\limits_{i \in OBJ} C_{1,s,s',i} \cdot y_i + v_{1s,s'} \geq 1 \\ \sum\limits_{i \in OBJ} C_{2s,s',i} \cdot y_i + v_{2s,s'} \geq 1 \end{array}\right\} \forall (s,s') \in P_{s,s'} \\ \quad \left.\begin{array}{l} \underline{\delta_i} \geq Pen_{1s,s',i} \cdot w_{1s,s',i} \\ \bar{\delta}_i \geq Pen_{2s,s',i} \cdot w_{2s,s',i} \\ y_i + (1 - v_{1s,s'}) + w_{1s,s',i} \geq 1 \\ y_i + (1 - v_{2s,s'}) + w_{2s,s',i} \geq 1 \\ (1 - y_i) + (1 - w_{1s,s',i}) \geq 1 \\ (1 - y_i) + (1 - w_{2s,s',i}) \geq 1 \\ v_{1s,s'} + 1 - w_{1s,s',i} \geq 1 \\ v_{2s,s'} + 1 - w_{2s,s',i} \geq 1 \end{array}\right\} \forall (s,s') \in P_{s,s'},\ \forall i \in OBJ \\ \quad y_i, v_{1,s,s'}, v_{2,s,s'} \in \{0,1\} \end{array}$$

(Model 3.1)

As an example to illustrate the difference between models, we go back to an example presented in the previous work (Vázquez et al., 2018), as example 3. The data is shown in Table 1.

Table 1: Data for norm 1 example

$x_{s,i}$	**i_1**	**i_2**	**i_3**	**i_4**	**i_5**	**i_6**	**i_7**	**i_8**	**i_9**
s_1	0.000	0.400	0.714	0.880	0.429	0.400	0.700	0.800	0.600
s_2	0.333	0.000	0.143	0.800	0.250	0.150	0.500	0.400	0.500
s_3	0.167	0.700	0.000	0.600	0.000	0.000	0.000	0.200	0.400
s_4	1.000	0.850	1.000	0.000	1.000	0.100	1.000	1.000	1.000
s_5	0.500	1.000	0.400	1.000	0.100	1.000	0.800	0.000	0.900

Both Model 3 and Model 3.1 are run with these data, using a binary cut strategy to obtain different runs and see how the different norms behave. The results are shown in Table 2.

Table 2: Results of Model 3 and 3.1 for data in Table 1

Run #	OBJ'	Model 3 $z = 100 \cdot \delta$	Model 3.1 $z = 100 \cdot \sum_{i \in OBJ} \overline{\delta_i} \;/\; Nobj$
1	$\{i_2, i_4, i_8\}$	33.30%	11.10%
2	$\{i_2, i_4, i_5\}$	33.30%	17.78%
3	$\{i_2, i_3, i_4\}$	40.00%	29.43%
4	$\{i_2, i_6, i_8\}$	60.00%	31.10%
5	$\{i_4, i_5\}$	70.00%	61.65%
6	$\{i_4, i_8\}$	70.00%	51.65%
7	$\{i_3, i_4\}$	70.00%	79.15%
8	$\{i_6, i_8\}$	70.00%	81.65%

As a result, for subsets that were treated as equally good with the previous model, now we can see a difference among them in order to choose one with preference to another.

3.2 Non-redundancy ratio

While the utilization of a norm 1 can refine the measure of the modifications introduced in the dominance structure of a system when an objective is removed, it still does not provide a completely faithful information of the amount of Pareto solutions that stop being Pareto optimal. This is due to the fact that the mere concept of δ-error does include an infinity norm, even if it is only considered objective per objective, as in the norm 1 model.

As such, either if we want to maintain the minimum maximum δ-error concept, which is utilized in both Model 2 and 3, or the norm 1 of maximum δ-error per objective, which is utilized in Model 3.1, it is still possible that the dominance structure is broken only by a reduced number of solutions. This is more common in cases in which the number of solutions is much larger than the number of objectives which is usually the case. To control this undesirable effect we introduce the concept of non-redundancy ratio.

It is defined as the percentage of the Pareto solutions that are maintained for a subset of up to a determined size of non-redundant objectives. It is shown in Eq.(5).

$$NR\ ratio = \frac{S'}{S_0} \cdot 100 \tag{5}$$

Where S' is the number of solutions that remain as Pareto solutions for a δ-error = 0 when maintaining a certain subset of objectives, and S_0 is the initial number of Pareto solutions of the system.

In the case treated in the theoretical foundations section (Figure 1), the subsets $\{f_1, f_3\}, \{f_2, f_3\}$ are non-redundant with a non-redundancy ratio of 100%. This means that in the whole space of initial solutions $\{A, B\}$, those subsets are able to reduce the number of objectives without incurring a δ-error. As an example to illustrate this concept, we observe the data represented in Figure 2, where we cannot remove any objective without incurring a δ-error. If we remove the first objective, D stops being Pareto optimal. If we remove the second, C stops being Pareto optimal. And if we remove the third one, A stops being Pareto optimal.

In this case, it is obvious that all 3 objectives are important. But sometimes, we have a high number of solutions and a much smaller number of objectives, and the dominance structure is broken only by a couple of solutions.

< Fig.2 >

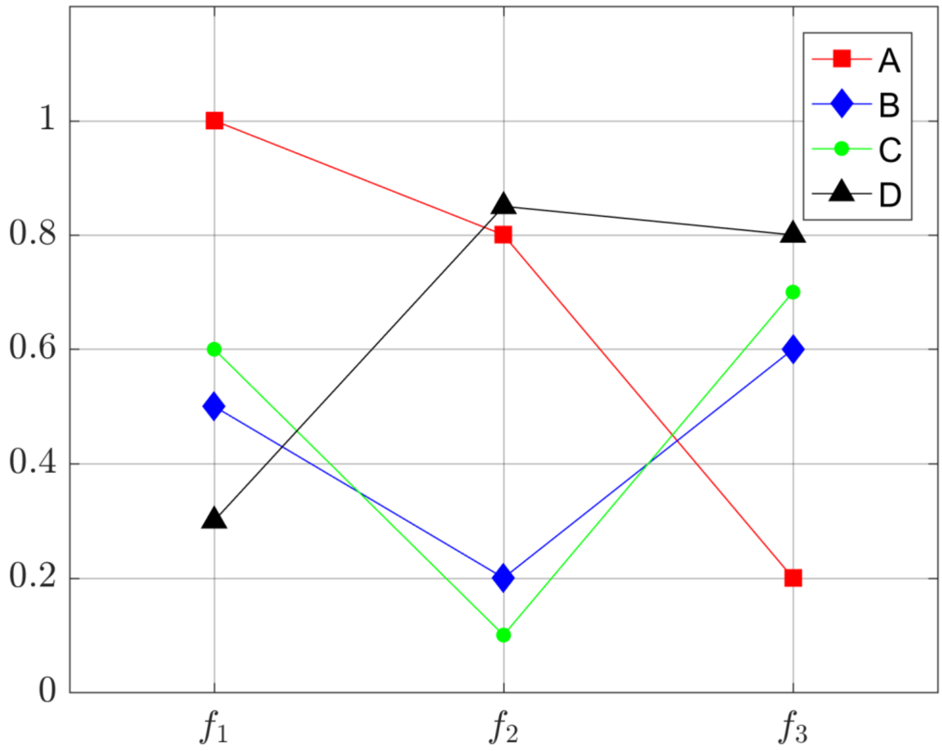

Figure 2: Parallel plot example for a case with 4 Pareto solutions and 3 objectives

The idea behind the non-redundancy ratio is as follows. If we remove solution D, it would be enough with the objectives f_2 and f_3 to maintain completely the dominance structure. Likewise, if we remove solution A, it would be enough with objectives f_1 and f_2 to maintain the dominance structure. As such, it is said that the subsets of objectives $\{f_2, f_3\}$ and $\{f_1, f_2\}$ are non-redundant with a ratio of 75%. Or, that the objectives f_1 and f_3 respectively are redundant with a ratio of 75%.

In order to account for the possibility of removing solutions, instead of removing objectives, which was the desired outcome up until now, the model must be modified. We introduce a new binary variable to the model, z_s, that will takes the value 1 when solution s is maintained and 0 otherwise. The new objective function is also modified, as shown in Eq.(6).

$$\sum_s z_s - \lambda \sum_i y_i \tag{6}$$

Where λ is a parameter with a small value, which ensures that when maximizing that objective function, the method will account for the minimum allowable number of objectives. The algorithm would work as well without this penalization expression, but then it would not have any reason to try to provide the minimum number of objectives as well as the maximum number of Pareto solutions maintained.

The constraints of Model 1 must also be modified as shown in Eq.(7), allowing them to break instantly its need for fulfillment if any of the two solutions being compared is not being maintained.

$$\begin{aligned} &\sum_{i \in OBJ} C_{1,s,s',i}\ y_i + (1 - z_s) + (1 - z_{s'}) \geq 1 \quad \forall (s,s') \in P_{s,s'} \\ &\sum_{i \in OBJ} C_{2,s,s',i}\ y_i + (1 - z_s) + (1 - z_{s'}) \geq 1 \quad \forall (s,s') \in P_{s,s'} \end{aligned} \tag{7}$$

We need as well to introduce a new constraint that will limit the number of objectives to the number that we want to maintain. We will call that parameter the same as in previous models, $Nobj$. In the previous example, $Nobj = 2$. The constraint is shown in Eq.(8).

$$\sum_i y_i \leq Nobj \tag{8}$$

As such, the model is defined as Model 4.

$$\max \quad \sum_s z_s - \lambda \sum_i y_i$$

$$s.t. \quad \sum_{i \in OBJ} C_{1,s,s',i} \; y_i + (1 - z_s) + (1 - z_{s'}) \geq 1 \quad \forall (s,s') \in P_{s,s'}$$

(Model 4)

$$\sum_{i \in OBJ} C_{2,s,s',i} \; y_i + (1 - z_s) + (1 - z_{s'}) \geq 1 \quad \forall (s,s') \in P_{s,s'}$$

$$\sum_i y_i \leq Nobj$$

$$y_i, z_s \in \{0,1\}$$

And, as it is obvious, $\sum_s z_s = S'$. By using this model, we can obtain the optimum NR ratio of a system.

In order to further exemplify this, we use again the data of Table 1. The results are shown in Table 3.

Table 3: Results of Model 3 and Model 4 for data in Table 1

Run #	OBJ'	Model 3 $z = 100 \cdot \delta$	Model 4 NR-ratio
1	$\{i_2, i_4, i_8\}$	33.30%	80%
2	$\{i_2, i_4, i_5\}$	33.30%	80%
3	$\{i_2, i_3, i_4\}$	40.00%	60%
4	$\{i_2, i_6, i_8\}$	60.00%	60%
5	$\{i_4, i_5\}$	70.00%	60%
6	$\{i_4, i_8\}$	70.00%	60%
7	$\{i_3, i_4\}$	70.00%	60%
8	$\{i_6, i_8\}$	70.00%	60%

We can see that the NR-ratio provides useful information. If we are considering the δ-error, we could argue that between $\{i_2, i_4, i_5\}$ and $\{i_2, i_3, i_4\}$ there is only a difference of a ~7 % in the δ-error so, if we wanted for some reason to keep i_3, we could justify its presence with that minimum deviation of the optimum subset of size 3. Thanks to the NR-ratio, we can see that the impact when choosing the later subset is more than just a 7 % in δ-error, since more solutions stop being Pareto optimal. Even so, in this case it can be seen that the NR-ratio is not very sensitive. This is due to the fact that the number of solutions is too small to make a difference in this system. In the following paragraphs, with the different case studies, it can be seen its utility as complementary measurement to the δ-error once the number of solutions becomes considerably higher than the number of objectives.

4. Comparison with the Principal Component Analysis (PCA) and Deb's algorithm

The Principal Component Analysis is a statistical procedure based upon performing an orthogonal transformation to convert a set of points, which are expected to have a correlation between them, to a series of uncorrelated linear variables. Those variables are called principal components.

The method provides an equal number of principal components as of variables, but it is normal to consider a threshold of *explained variance*. This threshold tends to be around +95%, so the number of principal components maintained is lower than the initial number of variables, otherwise the technique is not useful.

While this reduces the dimensionality of the problem in a majority of cases, due to the fact that the number of principal components maintained will be lower than the initial number of variables, it does not ease the calculations needed in the problem resolution. Each principal component will be formed by **all** the variables present at the initial problem. This meaning, that even though you are reducing the objectives to a lesser number of pseudo-objectives, you still have to calculate every single objective of the initial set.

Deb and Saxena Kumar (2005) developed a series of guidelines that allowed the user to remove objectives in the main space of objectives based on the results of the principal components. Those guidelines are resumed as:

1. Retain the amount of Principal Components that are over a certain threshold in accumulated explained variance.
2. For those Principal Components, if the eigenvalue is less than 0.1, add to the subset of objectives the objective with the higher absolute value.
3. If the eigenvalue is more than 0.1:
    a. If all the components of the principal component are positive, add to the subset of objectives the objective corresponding to the highest value.

b. If all are negative, add all the objectives to the subset of objectives.

c. If none of the previous two cases apply:

i. If the most positive element is smaller than the 90% of the absolute value of the most negative one, add the most negative objective to the subset of objectives.

ii. If the most positive is bigger than the 90% of the absolute value of the most negative one, but smaller than the absolute value of the most negative one, add both the most negative and most positive to the subset of objectives.

iii. If the absolute value of the most negative is bigger than the 80% of the most positive, but smaller than the most positive, add to the subset of objectives both the most negative and most positive.

iv. If none of the previous cases apply, add to the subset of objectives the most positive one.

4. If the subset of objectives differs from the original set of objectives, make the subset the new original set and repeat the procedure.

In the next section, the case studies presentation, we use both our models and the PCA combined with Deb & Saxena's algorithm, in order to compare the results obtained.

5. Case studies

5.1 First case study

For the first case study, the data are extracted from the works of Pozo et al. (2012). It is based on an existing supply chain established in Europe which comprises one plant and one warehouse, with four different markets and six different technologies to consider. While the entire optimization solution is not reproduced here, some of the resultant Pareto solutions are the only data needed. These Pareto solutions are shown in Table 4.

Table 4: Data of the first case study. 4 objectives and 16 solutions

Solution	f_1	f_2	f_3	f_4
s_1	-103,772	22,530,227	104,242,733	359,642,956
s_2	-113,850	22,889,709	106,029,789	365,276,235
s_3	-114,954	22,928,460	106,233,138	369,279,030
s_4	-114,980	22,930,280	106,242,154	369,309,474
s_5	-117,401	23,180,056	107,474,994	370,909,514
s_6	-119,897	23,330,333	108,220,586	376,260,912
s_7	-119,903	23,332,064	108,223,542	376,391,893
s_8	-120,158	23,450,603	108,787,926	376,542,793
s_9	-122,652	23,730,386	110,109,661	382,176,072
s_{10}	-122,681	23,730,386	110,103,519	382,300,057
s_{11}	-122,679	23,749,505	110,213,947	382,176,072
s_{12}	-122,781	23,752,498	110,213,947	382,505,703
s_{13}	-122,682	23,752,871	110,232,303	382,176,072
s_{14}	-123,716	24,130,438	112,147,778	385,082,154
s_{15}	-123,739	24,141,495	112,204,351	385,157,527
s_{16}	-124,284	24,530,491	114,194,755	387,809,352

Where f_1 refers to the economic objective of the Net Present Value in dollars. We want to maximize this objective. Since our models work for minimization problems, the sign is changed, as it is already shown in the table. The objectives f_2 to f_4 refer to environmental objectives; Ecosystem Quality (EQ), Human Health (HH) and Damage to Natural Resources (NR) respectively. Obviously, these are to be minimized.

5.1.1 Model 1 and Model 4

After running our Model 1, the non-redundant subsets of objectives are shown in Table 5. We can see that the cardinality of the minimum subset is 3.

Table 5: Results of Model 1 for the first case study

Subsets	Objectives
OBJ'_1	$\{f_1, f_2, f_4\}$
OBJ'_2	$\{f_1, f_3, f_4\}$

We now apply our model 4 for $N = 2$. We use a binary cut strategy in order to obtain the rest of the solutions in order to compare them. The results are shown in Table 6.

Table 6: Results of Model 4 for the first case study. Nobj = 2

Number of solutions removed	Objectives maintained	NR ratio	δ-error
2	$\{f_1, f_4\}, \{f_1, f_3\}, \{f_1, f_2\}$	87.5 %	1.29 %, 1.17 %, 1.17 %
11	$\{f_2, f_4\}, \{f_3, f_4\}$	25.0 %	100 %, 100 %
14	$\{f_2, f_3\}$	12.5 %	100 %

It is clear that the number of objectives can be reduced to two without an important loss in the dominance structure. A ratio of non-redundancy of the subsets of ~ 90% is maintained, with a δ-error rounding the 1%. It is important to note that Model 4 counts the three first subsets as equally good, while the δ-error differs slightly. Thus, both variables, NR-ratio and δ-error, must be checked in order to obtain the best subset of objectives.

5.1.2 PCA

If we perform a Principal Component Analysis, we obtain the results showcased in Table 7.

Table 7: PCA results for the first case study. 4 objectives

Objective	PC1	PC2	PC3	PC4

f_1	-0.4871	0.7700	0.4120	-0.0116
f_2	0.5033	0.4331	-0.1941	0.7221
f_3	0.5026	0.4486	-0.2638	-0.6903
f_4	0.5068	-0.1350	0.8503	-0.0437
Explained (%)	95.9235	3.6705	0.4045	0.0015

With a threshold of a 95% the first principal component would suffice to explain correctly the data. If we first reduce the dimensionality of the problem to three objectives, for example by using the first subset of objectives shown in Table 5, the number of PC remains the same, but this is calculated using only those three objectives, simplifying the calculations. The results are shown in Table 8.

Table 8: PCA results for the second case study. First subset of objectives.

Objective	**PC1**	**PC2**	**PC3**
f_1	-0.5715	0.7337	0.3675
f_2	0.5738	0.6775	-0.4602
f_4	0.5866	0.0522	0.8082
Explained (%)	95.9468	3.5852	0.4680

In this case study, the advantage is not clearly seen since the difficulty of calculating three objective versus four objectives seems small. In the second case study, the gap between the raw PCA and the PCA after reducing objectives is more noticeable.

5.1.3 PCA + Deb's algorithm

With a threshold of 95%, only the first principal component remains. Looking at the values in Table 7, we note, based on Deb's algorithm, that we must maintain both the most positive and the most negative one. Thus, in a first step of the algorithm, the subset of objectives is reduced to $\{f_1, f_4\}$. After reducing the space of objectives and obtaining new Pareto points, the results of Deb's algorithm show that the number of objectives cannot be further reduced, as the original article notes (Pozo et al., 2012).

As such, the result is the same that we obtained with our method, but it only obtains one of the subset of solutions, and does not provide any quantifiable error when doing so. Even so, it is not even the best subset of objectives in regards to the δ-error, considering that both the subset $\{f_1, f_3\}$ and the subset $\{f_1, f_2\}$ provide a smaller maximum δ-error.

5.2 Second case study

Considering the data found in the case study from Carreras et al. (2016), where the objective is to optimize the construction of a building considering both economic and environmental objectives. It has a space of solutions of 7776 solutions, with 12 different objectives. From those objectives, the twelfth one is an aggregated environmental index. It is interesting to study what happens whenever the aggregated index is taken into account, and what would mean to remove it from the study. Thus, the models are applied to both to the case that the initial 12 objectives are maintained, i.e., considering the aggregated environmental metric, and to the case when only 11 objectives, without the aggregated metric, are considered.

5.2.1 Model 1 and Model 4

From those 7776 solutions, a subspace of 200, equally spaced solutions is chosen. After applying our Model 1, the subsets of non-redundant objectives are shown in Table 9.

Table 9: Results of Model 1 for the second case study. 12 objectives

Subsets	Objectives	Subsets	Objectives
OBJ'_1	$\{f_1, f_2, f_5\}$	OBJ'_{10}	$\{f_1, f_2, f_{10}\}$
OBJ'_2	$\{f_1, f_{10}, f_{12}\}$	OBJ'_{11}	$\{f_1, f_3, f_{10}\}$
OBJ'_3	$\{f_1, f_5, f_{11}\}$	OBJ'_{12}	$\{f_1, f_4, f_{10}\}$
OBJ'_4	$\{f_1, f_5, f_9\}$	OBJ'_{13}	$\{f_1, f_7, f_{10}\}$

OBJ'_5	$\{f_1, f_5, f_8\}$	OBJ'_{14}	$\{f_1, f_9, f_{10}\}$
OBJ'_6	$\{f_1, f_5, f_7\}$	OBJ'_{15}	$\{f_1, f_{10}, f_{11}\}$
OBJ'_7	$\{f_1, f_5, f_6\}$	OBJ'_{16}	$\{f_1, f_6, f_{10}\}$
OBJ'_8	$\{f_1, f_4, f_5\}$	OBJ'_{17}	$\{f_1, f_5, f_{12}\}$
OBJ'_9	$\{f_1, f_3, f_5\}$	OBJ'_{18}	$\{f_1, f_8, f_{10}\}$

We apply now our Model 4 to the same data with $N = 2$. The results are shown in Table 10.

Table 10: Results of Model 4 for 12 objectives and Nobj = 2

Number of solutions removed	**Objectives maintained**	**NR ratio**	**δ-error**
1	$\{f_1, f_{12}\}$	99.5 %	24.01 %

As shown, we can practically say that these two objectives are non-redundant in regards of the non-redundancy ratio. However, the δ-error must be taken into account as well.

An important feature of this model is that it is much faster than the third model, as shown in Table 11.

Table 11: Comparison of the computational time for Models 4 and 3 with 12 objectives

Model 4: NR-ratio problem			
Blocks of equations :	4	Single equations :	39,802
Blocks of variables :	3	Single variables :	213
Non-zero elements :	318,625	Discrete variables :	212
Time (s) :	< 1		
Model 3: k-EMOSS problem			
Blocks of equations :	12	Single equations :	1,950,202
Blocks of variables :	7	Single variables :	517,414
Non-zero elements :	4,378,014	Discrete variables :	477,612
Time (s) :	≈ 4000		

All the calculations were performed with a PC using Windows 7 Professional 64-bits as OS, with an Intel® Core ™ i7-4790 CPU of 3.60 GHz and 8 Gb of RAM. The optimization was performed in GAMS using the solver Cplex.

We repeat the procedure without taking into account the aggregated environmental objective. The results of applying Model 1 are shown in Table 12.

Table 12: Results of Model 4 for the second case study. 11 objectives, Nobj = 2

SUBSETS	OBJECTIVES	SUBSETS	OBJECTIVES
OBJ'_1	$\{f_1, f_2, f_5\}$	OBJ'_9	$\{f_1, f_2, f_{10}\}$
OBJ'_2	$\{f_1, f_{10}, f_{11}\}$	OBJ'_{10}	$\{f_1, f_3, f_{10}\}$
OBJ'_3	$\{f_1, f_5, f_9\}$	OBJ'_{11}	$\{f_1, f_4, f_{10}\}$
OBJ'_4	$\{f_1, f_5, f_8\}$	OBJ'_{12}	$\{f_1, f_7, f_{10}\}$
OBJ'_5	$\{f_1, f_5, f_7\}$	OBJ'_{13}	$\{f_1, f_9, f_{10}\}$
OBJ'_6	$\{f_1, f_5, f_6\}$	OBJ'_{14}	$\{f_1, f_6, f_{10}\}$
OBJ'_7	$\{f_1, f_4, f_5\}$	OBJ'_{15}	$\{f_1, f_5, f_{11}\}$
OBJ'_8	$\{f_1, f_3, f_5\}$	OBJ'_{16}	$\{f_1, f_8, f_{10}\}$

The results of Model 4 are shown in Table 13.

Table 13: Results of Model 4 for 11 objectives and Nobj = 2

Number of solutions removed	Objectives maintained	NR ratio	δ-error

133	$\{f_1, f_{10}\}$	33.5 %	60.76 %

Both these parameters, δ-error and non-redundancy rank, are needed in order to fully understand how removing objectives affects the system. In this case, both the NR-ratio and δ-error get worse on the 11 objectives case, but this is not the only outcome possible. If there exist an objective that can be removed by removing only one solution, that is the solution that Model 4 would reach, while if there is another objective that by removing it removes 3 solutions, but with a smaller value of δ-error, that one is the solution that Model 3 would aim to. It can also be seen that when we maintain two objectives, if we consider the aggregated objective (f_{12}) the dominance structure is much better maintained than in this case, where we do not consider it.

5.2.2 PCA

Now, let's compare the results from our models with the results from the PCA. With 200 solutions and 12 objectives, the results of performing a PCA to the whole space are shown in Table 14, taking only into account the two first principal components, which explain practically completely the whole data.

Table 14: PCA results for the second case study. 12 objectives

Objective	**PC1**	**PC2**
f_1	-0.2986	-0.2077
f_2	0.3086	-0.0300
f_3	0.2983	-0.2106
f_4	0.2953	-0.2380
f_5	0.1614	0.6931
f_6	0.3075	0.0755
f_7	0.2872	-0.2989
f_8	0.2895	-0.2828

f_9	0.3079	0.0629
f_{10}	0.2920	0.3313
f_{11}	0.2986	-0.2069
f_{12}	0.2986	0.2075
Explained (%)	87.39	12.60

While the PCA reduces effectively the space to a two dimensional one, there must be noted that each of those principal components have as many coefficients as original objectives. This can make the calculations still very difficult to perform.

If we first reduce the set of objectives to a subset with zero δ-error, for example $OBJ_1 : \{f_1, f_2, f_5\}$, and then perform the PCA, we obtain the results showcased in Table 15.

Table 15: PCA results for the second case study. First subset of objectives.

Objective	**PC1**	**PC2**
f_1	0.6328	0.1577
f_2	-0.5840	-0.5456
f_5	-0.5084	0.8231
Explained (%)	82	18

It is also reduced to two principal components, but now it has the advantage of only needing three functions to calculate for each principal component, simplifying greatly the weight of the calculations. This behavior is analogous for when not considering the aggregated environmental metric.

As such, doing an initial dimensionality reduction can help immensely the PCA of the system, providing easier to calculate principal components. The PCA can as well help reduce the dimensionality space to a better representable one.

5.2.3 PCA + Deb's algorithm

With a threshold of 95%, both the first and the second principal components remain. Looking at the values in Table 14, we note, based on Deb's algorithm, that we must maintain both the most positive and the most negative one. Thus, in a first step of the algorithm, from the first principal component we must maintain the objectives f_1, f_9 and from the second principal component we must maintain f_5. After maintaining those, repeating the PCA, we obtain the data shown in Table 16.

Table 16: Second PCA for the second case study. Reduced objectives. Deb's algorithm

Objective	**PC1**	**PC2**
f_1	0.6195	0.2315
f_5	-0.5159	0.8403
f_9	-0.5917	-0.4902
Explained (%)	84.73	15.27

If we repeat the algorithm, now from the first PCA we maintain f_1, f_9, while from the second we must maintain f_5. As such, the algorithm cannot reduce any more the number of objectives.

This result was already obtained by our method, as well as multiple others, for a δ-error of 0.

6. Conclusions

A slight modification of the previous model, where the δ-error is assessed by a norm 1 of the different objectives is presented, which allows to further classify different subset of objectives that may seem equally good when considering only the maximum δ-error of the system. The model includes a normalization to maintain the δ-error inside a 0 – 100% limit, but it is recommended to perform the optimization without the normalization and then normalize the results if desired.

A new model has been developed to account for the amount of solutions that stop being Pareto optimal when removing objectives. The concept of non-redundancy ratio is introduced as this value, and proven its utility with the case studies. Ideally, the optimal subset would have the highest NR-ratio and the lowest δ-error. They are certainly related by the structure of the problem, but as shown in the results, both must be taken into account in order to completely classify a subset of objectives. This NR-ratio is especially effective when we are not sure of the bounds of each objective, since it is not sensible to them, while the δ-error is. Besides, the fact that the NR-ratio is especially sensitive to the number of solutions in the problem, while the δ-error, both in its infinity norm and norm 1 form, is especially sensitive to the number of objectives, as showcased by the examples and case studies, reinforces the complementarity between these concepts and the need of considering both when the objective is to completely study a dominance structure among different objectives.

Our models are then compared with another well-known objective reduction technique, this being the combination of the Principal Component Analysis and the algorithm from Deb & Saxena. It is being shown that our models are able to provide a quantifiable error when reducing objectives, while Deb's algorithm uses a heuristic approach that provides a solution that may or may not have an intrinsic error. As such, when reducing objectives using Deb's algorithm, we cannot be sure of the reduction not being the best one for the process.

While the PCA cannot reduce the number of objectives on its own, since it only aggregates them in weighted terms which allow for easier representation of data. If we perform a PCA of a set of data, the dimensionality will be reduced in the sense that the number of principal components maintained, for a given threshold, will be smaller than the initial number of objectives. However, these principal components require the total number of initial objectives to be computed. If on the other hand we start by using our methodology of objective reduction, and then we perform the PCA, we obtain the same benefit of reducing the dimensionality in the principal component space, where the reduced number of principal components now are not calculated operating with all the initial objectives, but only with the reduced subset of objectives maintained.

Acknowledgments

The authors acknowledge financial support from the Spanish"Ministerio de Economía, Industria y Competitividad" (CTQ2016-77968-C3-2-P, AEI/FEDER, UE).

Nomenclature

SETS

F, OBJ	Set of objectives indexed by k, i
OBJ'	Set of reduced objectives indexed by i
X, S	Set of solutions indexed by x, s
S'	Set of reduced solutions indexed by s
P	Subset of pairs of S indexed by (s, s') where $s < s'$

DATA

$x_{s,i}$	Value of the solution s in the objective i

PARAMETERS

$C_{1,s,s',i}, C_{2,s,s',i}$	Coefficients which have either a value of 1 or 0, depending on the data
$Pen_{1,s,s',i}, Pen_{2,s,s',i}$	Measurement of the error between pairs of solutions when one objective is removed
$Nobj$	Number of objectives to maintain in Model 3, 4

VARIABLES

y_i	Binary variable that takes the value 1 when objective i is chosen and 0 otherwise
z_s	Binary variable that takes the value 1 when solution s is maintained and 0 otherwise
$v_{1,s,s'}, v_{2,s,s'}$	Binary variables that takes the value 1 when the constraints of Model 1 are to be broken
$w_{1,s,s'}, w_{2,s,s'}$	Pseudo-binary variables that help formulate the logical constraints
δ	Value of the maximum δ-error of the system
$\overline{\delta_i}$	Value of the maximum δ-error of an objective i

References

Aamodt, A., & Plaza, E. (1994). Case-based reasoning: foundational issues, methodological variations, and system approaches. AI Commun., 7, 39-59.

Banker, R. D., Charnes, A., & Cooper, W. W. (1984). Some Models for Estimating Technical and Scale Inefficiencies in Data Envelopment Analysis. Management Science, 30, 1078-1092.

Brockhoff, D., & Zitzler, E. (2006a). Are All Objectives Necessary? On Dimensionality Reduction in Evolutionary Multiobjective Optimization. TIK - Report.

Brockhoff, D., & Zitzler, E. (2006b). Dimensionality Reduction in Multiobjective Optimization with (Partial) Dominance Structure Preservation: Generalized Minimum Objective Subset Problems. TIK - Report, 247.

Brockhoff, D., & Zitzler, E. (2006c). On Objective Conflicts and Objective Reduction in Multiple Criteria Optimization. TIK - Report, 243.

Brockhoff, D., & Zitzler, E. (2009). Objective reduction in evolutionary multiobjective optimization: theory and applications. Evolutionary Computation, 17, 135-166.

Carreras, J., Pozo, C., Boer, D., Guillén-Gosálbez, G., Caballero, J. A., Ruiz-Femenia, R., & Jiménez, L. (2016). Systematic approach for the life cycle multi-objective optimization of buildings combining objective reduction and surrogate modeling. Energy and Buildings, 130, 506-518.

Das, I. (1999). A preference ordering among various Pareto optimal alternatives. Structural Optimization, 18, 30-35.

Deb, K., & Saxena Kumar, D. (2005). On Finding Pareto-Optimal Solutions Through Dimensionality Reduction for Certain Large-Dimensional Multi-Objective Optimization Problems. KanGAL Report, 200511.

Goedkoop, M. J., Heijungs, R., De Schyver, A., Struijs, J., & Van Zelm, R. (2009). ReCiPe 2008, A life cycle impact assessment method which comprises harmonised category indicators at the midpoint and the endpoint level.

Guillén-Gosálbez, G. (2011). A novel MILP-based objective reduction method for multi-objective optimization: Application to environmental problems. Computers & Chemical Engineering, 35, 1469-1477.

Haimes, Y. Y., Lasdon, L. S., & Wismer, D. A. (1971). On a Bicriterion Formulation of the Problems of Integrated System Identification and System Optimization. IEEE Transactions on Systems, Man, and Cybernetics.

Jolliet, O., Margni, M., Charles, R., Humbert, S., Payet, J., Rebitzer, G., & Rosenbaum, R. (2003). IMPACT 2002+: A new life cycle impact assessment methodology. The International Journal of Life Cycle Assessment, 8, 324.

Pearson, K. (1901). On Lines and Planes of Closest Fit to Systems of Points in Space. Philosophical Magazine, 2.

Pozo, C., Ruíz-Femenia, R., Caballero, J., Guillén-Gosálbez, G., & Jiménez, L. (2012). On the use of Principal Component Analysis for reducing the number of environmental objectives in multi-objective optimization: Application to the design of chemical supply chains. Chemical Engineering Science, 69, 146-158.

PRé-Consultants. (2000). The eco-indicator 99: A damage oriented method for life cycle impact assessment, methodology report and manual for designers. Amersfoort, The Netherlands: Pré Consultants.

Saaty, T. L. (1990). Decision Making for Leaders: The Analytic Hierarchy Process for Decisions in a Complex World: University of Pittsburgh.

Saaty, T. L. (2001). Decision Making with Dependence and Feedback: The Analytic Network Process : the Organization and Prioritization of Complexity: Rws Publications.

Vázquez, D., Fernández-Torres, M. J., Ruiz-Femenia, R., Jiménez, L., & Caballero, J. A. (2018). MILP method for objective reduction in multi-objective optimization. Computers & Chemical Engineering, 108, 382-394.

Zadeh, L. A. (1965). Fuzzy sets. Information and control, 8, 338-353.