A Model for the Decision-Maker Preferences in a Polymer Extrusion Process

Luciana Rocha Pedro[1], Ricardo Hiroshi Caldeira Takahashi[2], and António Gaspar-Cunha[3]

[1] COPPE, UFRJ, Rio de Janeiro, Brasil
lu.ufmg@gmail.com
[2] Department of Mathematics, UFMG, Belo Horizonte, Brasil ✉
taka@mat.ufmg.br
[3] Department of Polymer Engineering, Universidade do Minho, Guimarães, Portugal
agc@dep.uminho.pt

Abstract. The decision-making process is an important part of science in which specialists, called Decision-Makers (DM), apply their knowledge to make informed decisions. The NN-DM is a method developed to find a mathematical model for the DM in situations in which the preferences are represented by a utility function. This paper presents adaptations in the original NN-DM method to find a model for the DM's preferences applied in a polymer extrusion process. The DM's requirement is to fill a matrix expressing his preferences considering pairwise comparisons expressing ordinal relations only. Two multi-objective optimization problems are tested; each one with three estimates of different Pareto-optimal fronts. The adapted NN-DM method is able to provide a model which sorts the available solutions from the best to the worst one according to the DM's preferences.

Keywords: Multi-objective optimization problem · polymer extrusion process · artificial neural network · multi-criteria decision analysis · utility function.

1 Introduction

Most real-world optimization problems involve multiple objectives which have to be considered simultaneously. As these objectives are usually conflicting it may not be possible to find a single solution which is optimal with respect to all objectives. For obtaining only one solution a Decision-Maker (DM) has to make a choice regarding the importance of different criteria related to the optimization process [1]. Therefore, the final single solution of a Multi-Objective Optimization Problem (MOOP) results from the combined optimization and decision processes.

The importance of the decision-making process in a multi-objective environment is recognized by the number of recent publications. Deb et al. proposed an interactive Multi-Objective Evolutionary Algorithm (MOEA) based on progressively approximated value functions [2]. The DM's preference information is

captured progressively by constructing a value function based on a preference order defined by the DM by pairwise comparison between the current solutions. In an approach considering reference points Köksalan and Karahan developed the Interactive Territory-Defining Evolutionary Algorithm (iTDEA) [5] which creates territories around the solutions with sizes reflecting the DM's preferences. The iTDEA guides the search converging to the entire Pareto-optimal front with preferable regions highlighted by its density. The quality of the described methods concerning the correspondence between the produced solutions and the DM's preferences is assessed empirically.

The NN-DM method [7] is a procedure for constructing a mathematical model for the DM in situations in which the preferences are according to the Multi-Attribute Utility Theory (MAUT), that is, the preferences are represented by an underlying utility function $\mathcal{U}$. The DM is required to express his preferences by pairwise comparisons expressing ordinal relations only. The process of providing information is made *a posteriori* and the data is employed in constructing a model, denoted NN-DM model $\hat{\mathcal{U}}$, that represents the DM's preferences in a specific domain $\mathcal{D}$. The function $\hat{\mathcal{U}}$ can be repeatedly employed whenever the available solutions are within the domain $\mathcal{D}$ without further demand to the DM.

Adaptations in the NN-DM method are performed to enable it to construct a model for the DM's preferences in a real scenario. The application considered here is one important polymer processing technology: the single screw extrusion [3]. The process performance depends on three different parameters: the polymer properties, the system geometry, and the operating conditions. Two MOOPs are examined: mass output × power consumption and mass output × WATS. In each MOOP three sets of Pareto-optimal Front Estimates (PFE) are available considering different sets of decision variables: the operating conditions, the screw geometry, and a combination of both. In each scenario the resulting NN-DM model $\hat{\mathcal{U}}$ provides the sorting of solutions belonging to the PFE from the best to the worst one according to the DM's preferences.

This paper is organized as follows. In Section 2 the definitions of multi-objective optimization and multi-criteria decision-making problems are presented. In Section 3 the polymer extrusion process is explained and the available data is stated. In Section 4 the adaptations in the NN-DM method are introduced and an example of the original NN-DM method is shown. In Section 5 the resulting NN-DM models are established and tests with the available data illustrate the models' behavior. Section 6 discusses the obtained results and the work under development.

2 Problem Statement

2.1 Multi-objective Optimization

A Multi-Objective Optimization Problem (MOOP) is concerned with mathematical optimization problems involving more than one objective function to be optimized simultaneously. In a MOOP a set of different optimal solutions may

exist in which no single solution can be considered better than the others with respect to all the criteria. This set is called Pareto-optimal set and its image in the space of the objectives is called Pareto-optimal front, or just Pareto-front. In the absence of any additional preference information, none of the Pareto-optimal solutions can be said to be inferior when compared to any other solution, as they are superior in at least one criterion.

Solving a MOOP is often a difficult task since it involves conflicting criteria and usually several constraints are present. Due to these characteristics, specialized optimization algorithms are required to determine the set of solutions. Among these algorithms, Multi-Objective Evolutionary Algorithms (MOEAs) have become popular [8]. In the current application the Reduced Pareto Set Genetic Algorithm with Elitism (RPSGAe) [4, 3] is the MOEA selected to solve the polymer extrusion process.

RPSGAe is an algorithm based on the assignment of the fitness through a ranking function obtained employing a clustering algorithm. This optimization methodology has already been applied to the optimization of the operating conditions and to the design of screws for polymer extrusion. The results obtained by Gaspar-Cunha and Covas [4] showed that RPSGAe is able to find solutions with physical meaning in the proposed application.

2.2 Decision-Making Methodology

The selection of a single solution from a Pareto-front resultant from an optimization process requires information that may not be present in the objective functions. This information, expressing subjective preferences, must be introduced by a Decision-Maker (DM). The insertion of the DM's preferences in the optimization procedure allows the distinction among the solutions within a non-dominated set and, as a consequence, provides a ranking of the MOOP's solutions.

In this work the DM indicates preference relations (ordinal relations only) among simulated alternatives in the desired domain leading the NN-DM method to construct a model for the DM's preferences. The following basic elements are involved:

Set A of available alternatives This set is an estimate of the Pareto-front provided by RPSGAe which works as a problem instance of the multi-criteria decision-making problem. The set A is discrete and each element $a \in A$ corresponds to a solution located on the PFE.

Decision-Maker Each alternative possesses a *value* which is assigned by a Decision-Maker (DM) that formally corresponds to a utility function $\mathcal{U}$. The best alternative $x^* \in A$ is the one that maximizes the function $\mathcal{U}$ in the set A. It is assumed here that it is not possible to directly measure the values of $\mathcal{U}(x)$, for any alternative x. Only the ordinal information, provided by a preference function U, may be extracted from yes/no queries to the DM.

Set F of simulated alternatives This set is constructed to request information from the DM about the entire domain $\mathcal{D}$ in which the utility function $\mathcal{U}$ is being approximated; the alternatives on a Pareto-optimal front usually does not fully provide this kind of information.

3 Polymer Extrusion Process

3.1 Process Description

Single screw extrusion is an important polymer processing technology allowing the production of products such as pipes, film, profiles, and fibers. The main basic functions of a single screw extruder are: to transport the solid material from the hopper to the heated barrel zone; to melt the polymer; to homogenize and mix the melted polymer with the additives usually present; and to create the necessary pressure which enables the polymer to pass through the die at the desired output. Different polymers are characterized by properties such as: thermal (heat conduction coefficient, melting temperature, heat capacity, etc.), physical (friction coefficients, density, etc.), and rheological (which is a measure of the resistance of the polymer to the flow).

In industrial practice the polymer processing technology is employed in managing a single polymer whose properties change according to pressure and temperature. Therefore, a thermo-mechanical environment is developed in which the polymer passes through different thermal and physical states. Figure 1 illustrates a simple extruder with a conventional screw with five geometrical zones:

(i) solids conveying in the hopper: the solids are fed into the hopper in which, by action of gravity, are transported inside the barrel;

(ii) solids conveying in the screw: by action of the screw rotation and due to the friction between the screw and barrel walls the solid polymer is pressurized and a solid bed is formed and, simultaneously, the polymer is transported to the heated barrel zone;

(iii) delay zone: due the the heat generated by friction and the heat conducted from the barrel a melt film is formed;

(iv) melting zone: a specific melting mechanism, characterized by the existence of a melt pool and melt films around the solid bed, is developed;

(v) conveying zone: the polymer is pressurized and it is transported to the die.

The modeling of the polymer processing technology involves the linkage of all those functional zones adopting the appropriate boundary conditions. The process performance depends on different type of parameters (polymer properties, system geometry, and operating conditions) which can be characterized by
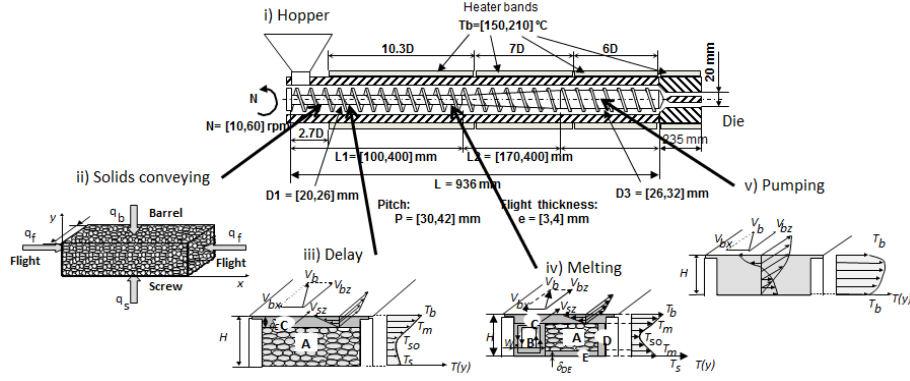
Fig. 1. Thermo-mechanical functional process developed in a single screw extruder

the mass output of the machine ($\mathbf{Q}$), the average melt temperature of the polymer at die exit ($\mathbf{T_{melt}}$), the power consumption required to rotate the screw ($\mathbf{P}$), the capacity of pressure generation ($\mathbf{P_{max}}$), the length of screw required to melt the polymer ($\mathbf{L_{melt}}$), and the degree of mixing quantified by the average of the deformation induced, denoted WATS ($\mathbf{W}$). Those are the common objectives considered in the definition of the multi-objective optimization problem related to the polymer extrusion process. Further details of the modeling routine implemented can be found elsewhere [3].

3.2 Available Data

As the single screw extrusion is a computationally expensive multi-objective optimization problem this paper deals directly with estimates of different Pareto-optimal fronts obtained by the RPSGAe multi-objective optimization algorithm. The objectives considered in the multi-objective optimization problems are: the mass output of the machine ($\mathbf{Q}$), the power consumption required to rotate the screw ($\mathbf{P}$), and the mixing capacity measure by the average of deformation ($\mathbf{W}$). Two problems are then considered: mass output ($\mathbf{Q}$) $\times$ power consumption ($\mathbf{P}$) and mass output ($\mathbf{Q}$) $\times$ WATS ($\mathbf{W}$). In each problem three sets of PFE are available considering different decision variables. The first set considers the operating conditions given by $\mathbf{N}$, $\mathbf{T_{b1}}$, $\mathbf{T_{b2}}$, and $\mathbf{T_{b3}}$; the second set considers the geometry in which the variables are $\mathbf{L_1}$, $\mathbf{L_2}$, $\mathbf{D_1}$, $\mathbf{D_2}$, $\mathbf{Pitch}$, and $\mathbf{e}$; and, finally, in the third set both types of decision variables are considered. Table 1 resumes the information about the available PFE and Table 2 provides the objectives, aim of optimization, range of variation, and the partitions.

Figure 2 presents the available estimates of the Pareto-optimal fronts considering the problems mass output $\times$ power consumption ($\mathbf{QP}_1$, $\mathbf{QP}_2$, and $\mathbf{QP}_3$) and mass output $\times$ WATS ($\mathbf{QW}_1$, $\mathbf{QW}_2$, and $\mathbf{QW}_3$). The domain is established by the range of variation presented in Table 2.

Table 1. Multi-objective optimization problems in a single screw extrusion process

PFE	Objectives	Optimization Type	Decision Variables
$\mathbf{QP_1}$	$\mathbf{Q}$ and $\mathbf{P}$	Operating conditions	$\mathbf{N}$, $\mathbf{T_{b1}}$, $\mathbf{T_{b2}}$, $\mathbf{T_{b3}}$
$\mathbf{QP_2}$	$\mathbf{Q}$ and $\mathbf{P}$	Geometry	$\mathbf{L_1}$, $\mathbf{L_2}$, $\mathbf{D_1}$, $\mathbf{D_2}$, $\mathbf{Pitch}$, $\mathbf{e}$
$\mathbf{QP_3}$	$\mathbf{Q}$ and $\mathbf{P}$	Both	$\mathbf{N}$, $\mathbf{T_{b1}}$, $\mathbf{T_{b2}}$, $\mathbf{T_{b3}}$, $\mathbf{L_1}$, $\mathbf{L_2}$, $\mathbf{D_1}$, $\mathbf{D_2}$, $\mathbf{Pitch}$, $\mathbf{e}$
$\mathbf{QW_1}$	$\mathbf{Q}$ and $\mathbf{W}$	Operating conditions	$\mathbf{N}$, $\mathbf{T_{b1}}$, $\mathbf{T_{b2}}$, $\mathbf{T_{b3}}$
$\mathbf{QW_2}$	$\mathbf{Q}$ and $\mathbf{W}$	Geometry	$\mathbf{L_1}$, $\mathbf{L_2}$, $\mathbf{D_1}$, $\mathbf{D_2}$, $\mathbf{Pitch}$, $\mathbf{e}$
$\mathbf{QW_3}$	$\mathbf{Q}$ and $\mathbf{W}$	Both	$\mathbf{N}$, $\mathbf{T_{b1}}$, $\mathbf{T_{b2}}$, $\mathbf{T_{b3}}$, $\mathbf{L_1}$, $\mathbf{L_2}$, $\mathbf{D_1}$, $\mathbf{D_2}$, $\mathbf{Pitch}$, $\mathbf{e}$

Table 2. Objectives, aim of optimization, range of variation, and partitions

Objective	Aim of optimization	Range of variation	Partition
Mass output	Maximization	$[1, 20]$	$[f_{10}, f_{11}, f_{12}, f_{13}] = [1, 7, 14, 20]$
Power consumption	Minimization	$[0, 9200]$	$[f_{20}, f_{21}, f_{22}, f_{23}] = [0, 3067, 6134, 9200]$
WATS	Maximization	$[0, 1300]$	$[f_{30}, f_{31}, f_{32}, f_{33}] = [0, 434, 867, 1300]$

Mass output $\times$ Power consumption   Mass output $\times$ WATS
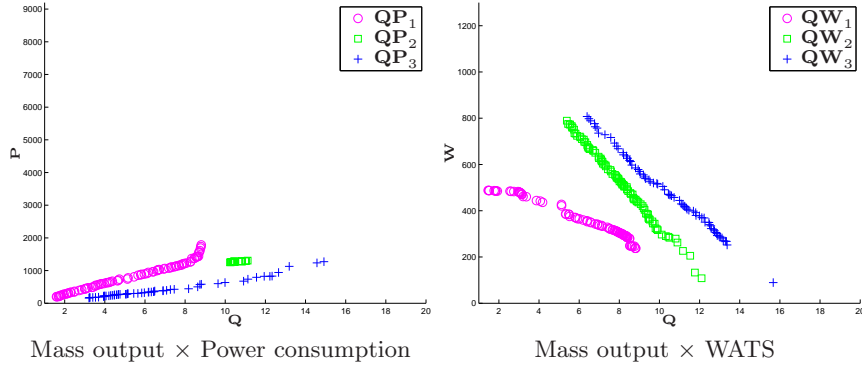
Fig. 2. Available estimates of the Pareto-optimal fronts

3.3 Interaction with the DM

A decision-making matrix $\mathcal{M}$ is a matrix filled by the DM to assist the NN-DM method in the construction of a model for the DM's preferences. Each element m_{ij} of $\mathcal{M}$ is defined as given in Equation 1.

$$\begin{cases} m_{i,j} = -1, & \text{if } a_i \text{ is preferable than } a_j; \\ m_{i,j} = 0, & \text{if } a_i \text{ and } a_j \text{ are equivalents;} \\ m_{i,j} = 1, & \text{if } a_j \text{ is preferable than } a_i. \end{cases} \tag{1}$$

Considering n the number of partitions in each dimension and m the number of objectives the total number of simulated alternatives is given by n^m. Therefore, the total number of pairwise comparisons is given by n^{2m} which corresponds to the number of entries of the decision-making matrix. The information required from the DM is reduced by dominance and comparisons between the same alternative (the matrix diagonal). The symmetry also develops an important role: given a utility function $\mathcal{U}$ and two alternatives a and b, $\mathcal{U}(a, b) = \mathcal{U}(b, a) = a$ or $\mathcal{U}(a, b) = \mathcal{U}(b, a) = b$. In both scenarios only one query is required from the DM and the entries in the decision-making matrix are symmetric values ± 1.

For exemplifying the whole described process consider a decision-making problem with two objective functions F_1 and F_2 whose aim is to minimize. The function F_1 is defined in the interval $[a_1, b_1]$ and the function F_2 is defined in the interval $[a_2, b_2]$. Assuming the number of partitions in each dimension of the grid established as 2 the interval partition matches with the interval extremes. Therefore, the decision-making process consists of 16 queries demanded by the combination of elements in the set $\{[a_1, a_2], [a_1, b_2], [b_1, a_2], [b_1, b_2]\}$.

The solutions to the queries s_i are divided into four groups:

Equivalence The solutions s_1, s_6, s_{11}, and s_{16} derive from queries made between the same alternative (the matrix diagonal). Therefore, the answer is zero since the solutions are equivalent.

Dominance The solutions s_2, s_3, s_4, s_8, and s_{12} are obtained considering the dominance, since $a_1 < a_2$, $b_1 < b_2$ and the aim of the optimization for both objectives is minimization.

Symmetry The solutions s_5, s_9, s_{13}, s_{14}, and s_{15} result from symmetry, since if the preferred alternative between a and b is, for example, a, the preferred alternative between b and a is also a.

Decision-Maker The solutions s_7 and s_{10} demand the DM's expertise. Considering that s_7 and s_{10} are provided from queries between the same alternatives only one query has to be presented to the DM.

Table 3a presents the unfilled decision-making matrix $\mathcal{M}$ for this example, with variables $s_i, i = 1 \ldots 16$, representing the entries. Table 3b shows the decision-making matrix partially filled by considering the equivalence, the dominance, and the symmetry among the alternatives. This matrix is then presented to the DM who needs to provide an answer to the remaining queries. In this example, only one query would be required from the DM.

In the real scenario considered here the number of partitions in each dimension of the grid is established as 4. This value provides enough information for the NN-DM method for constructing suitable NN-DM models for the DM's preferences without requiring demanding information from the DM.

Table 3. (a) Decision-making matrix $\mathcal{M}$ (b) Decision-making matrix $\mathcal{M}$ presented to the DM

$F_1 \times F_2$	$[a_1, a_2]$	$[a_1, b_2]$	$[b_1, a_2]$	$[b_1, b_2]$
$[a_1, a_2]$	s_1	s_2	s_3	s_4
$[a_1, b_2]$	s_5	s_6	s_7	s_8
$[b_1, a_2]$	s_9	s_{10}	s_{11}	s_{12}
$[b_1, b_2]$	s_{13}	s_{14}	s_{15}	s_{16}

$F_1 \times F_2$	$[a_1, a_2]$	$[a_1, b_2]$	$[b_1, a_2]$	$[b_1, b_2]$
$[a_1, a_2]$	0	1	1	1
$[a_1, b_2]$	-1	0	s_7	1
$[b_1, a_2]$	-1	s_{10}	0	1
$[b_1, b_2]$	-1	-1	-1	0

As each optimization problem is composed of two objective functions, there are 16 pairs of simulated alternatives which generate a total of 256 pairwise comparisons per problem. Excluding the comparison of pairs composed by the same alternatives (the matrix diagonal) and considering that $\mathcal{M}$ is anti-symmetric the resulting number of queries is given by 120. Among these 120 queries the dominance is applied considering the aim of optimization in each scenario leading to 84 solved queries. Therefore the DM had to answer to only 36 among those 256 queries in each optimization problem. The resulting matrix had been presented to the DM who had to choose the best alternative of each pair of simulated alternatives whose answer was not obtained by one of those described decision criteria. The decision-making matrices employed in estimating the DM's preferences in the polymer extrusion process are presented in Figures 4 and 5. The gray cells indicate the 36 positions the DM actually filled.

Table 4. Decision-making matrix: mass output ($\mathbf{Q}$) × power consumption ($\mathbf{P}$)

$\mathbf{Q} \times \mathbf{P}$	$[f_{10},f_{20}]$	$[f_{11},f_{20}]$	$[f_{12},f_{20}]$	$[f_{13},f_{20}]$	$[f_{10},f_{21}]$	$[f_{11},f_{21}]$	$[f_{12},f_{21}]$	$[f_{13},f_{21}]$	$[f_{10},f_{22}]$	$[f_{11},f_{22}]$	$[f_{12},f_{22}]$	$[f_{13},f_{22}]$	$[f_{10},f_{23}]$	$[f_{11},f_{23}]$	$[f_{12},f_{23}]$	$[f_{13},f_{23}]$
$[f_{10},f_{20}]$	0	1	1	1	-1	1	1	1	-1	1	1	1	-1	-1	-1	1
$[f_{11},f_{20}]$	-1	0	1	1	-1	-1	1	1	-1	-1	-1	1	-1	-1	-1	-1
$[f_{12},f_{20}]$	-1	-1	0	1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
$[f_{13},f_{20}]$	-1	-1	-1	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
$[f_{10},f_{21}]$	1	1	1	1	0	1	1	1	-1	1	1	1	-1	-1	-1	1
$[f_{11},f_{21}]$	-1	1	1	1	-1	0	1	1	-1	-1	1	1	-1	-1	-1	-1
$[f_{12},f_{21}]$	-1	-1	1	1	-1	-1	0	1	-1	-1	-1	1	-1	-1	-1	-1
$[f_{13},f_{21}]$	-1	-1	1	1	-1	-1	-1	0	-1	-1	-1	-1	-1	-1	-1	-1
$[f_{10},f_{22}]$	1	1	1	1	1	1	1	1	0	1	1	1	-1	-1	1	1
$[f_{11},f_{22}]$	-1	1	1	1	-1	1	1	1	-1	0	1	1	-1	-1	1	1
$[f_{12},f_{22}]$	-1	1	1	1	-1	-1	1	1	-1	-1	0	1	-1	-1	-1	1
$[f_{13},f_{22}]$	-1	-1	1	1	-1	-1	-1	1	-1	-1	-1	0	-1	-1	-1	-1
$[f_{10},f_{23}]$	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1
$[f_{11},f_{23}]$	1	1	1	1	1	1	1	1	1	1	1	1	-1	0	1	1
$[f_{12},f_{23}]$	1	1	1	1	1	1	1	1	-1	-1	1	1	-1	-1	0	1
$[f_{13},f_{23}]$	-1	1	1	1	-1	1	1	1	-1	-1	-1	1	-1	-1	-1	0

4 The Adapted NN-DM Methodology

This paper deals with a slight modification of the NN-DM method [7]. The NN-DM method is an algorithm developed to find a model, denoted NN-DM model $\hat{\mathcal{U}}$, which simulates the DM's preferences in situations in which these preferences are represented by a utility function $\mathcal{U}$. First, the domain $\mathcal{D}$ of the approximation

Table 5. Decision-making matrix: mass output ($\mathbf{Q}$) $\times$ WATS ($\mathbf{W}$)

$\mathbf{Q} \times \mathbf{W}$	$[f_{10}, f_{30}]$	$[f_{11}, f_{30}]$	$[f_{12}, f_{30}]$	$[f_{13}, f_{30}]$	$[f_{10}, f_{31}]$	$[f_{11}, f_{31}]$	$[f_{12}, f_{31}]$	$[f_{13}, f_{31}]$	$[f_{10}, f_{32}]$	$[f_{11}, f_{32}]$	$[f_{12}, f_{32}]$	$[f_{13}, f_{32}]$	$[f_{10}, f_{33}]$	$[f_{11}, f_{33}]$	$[f_{12}, f_{33}]$	$[f_{13}, f_{33}]$
$[f_{10}, f_{30}]$	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
$[f_{11}, f_{20}]$	-1	0	1	1	-1	-1	1	1	-1	-1	-1	1	-1	-1	-1	-1
$[f_{12}, f_{20}]$	-1	-1	0	1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
$[f_{13}, f_{20}]$	-1	-1	-1	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
$[f_{10}, f_{21}]$	1	1	1	1	0	1	1	1	-1	1	1	1	-1	-1	-1	1
$[f_{11}, f_{21}]$	-1	1	1	1	-1	0	1	1	-1	-1	1	1	-1	-1	-1	-1
$[f_{12}, f_{21}]$	-1	-1	1	1	-1	-1	0	1	-1	-1	-1	1	-1	-1	-1	-1
$[f_{13}, f_{21}]$	-1	-1	1	1	-1	-1	-1	0	-1	-1	-1	-1	-1	-1	-1	-1
$[f_{10}, f_{22}]$	1	1	1	1	1	1	1	1	0	1	1	1	-1	-1	1	1
$[f_{11}, f_{22}]$	-1	1	1	1	-1	1	1	1	-1	0	1	1	-1	-1	1	1
$[f_{12}, f_{22}]$	-1	1	1	1	-1	-1	1	1	-1	-1	0	1	-1	-1	-1	1
$[f_{13}, f_{22}]$	-1	-1	1	1	-1	-1	-1	1	-1	-1	-1	0	-1	-1	-1	-1
$[f_{10}, f_{23}]$	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1
$[f_{11}, f_{23}]$	1	1	1	1	1	1	1	1	1	1	1	1	-1	0	1	1
$[f_{12}, f_{23}]$	1	1	1	1	1	1	1	1	-1	-1	1	1	-1	-1	0	1
$[f_{13}, f_{23}]$	-1	1	1	1	-1	1	1	1	-1	-1	-1	1	-1	-1	-1	0

is established on the basis of the domain of the available alternatives A. Second, a partial ranking is built by answers to pairwise comparisons provided by the DM expressing ordinal relations only. Last, an artificial neural network is employed in approximating the partial ranking resulting in a model $\hat{\mathcal{U}}$ that has the same level sets of the DM's utility function $\mathcal{U}$. The NN-DM model $\hat{\mathcal{U}}$ is now able to represent the DM's preferences in alternatives belonging to the domain $\mathcal{D}$ without further queries to the DM.

The real DM considered here it is assumed to have preferences represented by a utility function $\mathcal{U}$ and hence by a NN-DM model $\hat{\mathcal{U}}$. However, the absence of an underlying utility function demands adaptations in some steps of the NN-DM method to consider the real DM. The interactions with the DM are also made in a different way: the DM has to fill a decision-making matrix regarding the unsolved queries (Section 3.3).

The original NN-DM method is divided into four steps. In the current work the domain $\mathcal{D}$ is previously provided by the DM. Thereby it is not necessary to establish the domain as the original **Step 1** has proposed. **Step 2** introduces the ranking of alternatives which is now built from a total sorting (the decision-making matrix). **Step 3** is unchanged, but additional changes are made in **Step 4** since the performance of the resulting model, assessed by the Kendall-Tau Distance (KTD) in the original method, is now evaluated by the DM himself. The original NN-DM method and the reported changes are better described next.

4.1 Step 1: Domain Establishment

In the NN-DM method the domain $\mathcal{D}$ is established from the available alternatives A. In the original model the alternatives A are employed in constructing a box whose values varies between the minimum and maximum values of the alternatives in each dimension problem. Into this domain a set of random simulated alternatives F is created and employed in building a ranking of alternatives and therefore the NN-DM model $\hat{\mathcal{U}}$.

In this paper the DM provides the decision-making domain which is employed in establishing the domain $\mathcal{D}$ of the model $\hat{\mathcal{U}}$. Into the domain $\mathcal{D}$ a grid of

simulated alternatives F is constructed to extract information about the DM's preferences. The grid is considered in an attempt to make the DM's analysis easier.

4.2 Step 2: Ranking Construction

The original NN-DM method builds a partial ranking $\mathcal{R}$ of the alternatives which assigns a scalar value to each alternative. Considering a set $\mathcal{A}$ with n alternatives, a subset with $p = \log n$ alternatives,[1] denoted *pivots*, is randomly constructed from the set $\mathcal{A}$. The pivots are sorted in ascending order of the DM's preferences and a rank is assigned to each pivot. Next the $n - p$ remaining alternatives are clustered into the classes defined by the $\log n$ pivots.

In an attempt to simplify this process to the real DM a decision-making matrix is constructed. The equivalence, the dominance, and the symmetry are first considered to take the decision in situations in which the answer is acquired without consulting the DM. The remaining queries are then presented to the DM as the decision-matrix $\mathcal{M}$ which captures the DM's preferences within the domain $\mathcal{D}$ and can be filled bu the DM in his own time.

Since the answers to all the queries are supplied by the matrix $\mathcal{M}$ a total ranking $\mathcal{R}$ is now available. Even knowing that the total sorting provides additional information the partial ranking is employed here since the DM's scale is unknown and the integer scale is inconvenient to the approximation technique. The partial ranking is built preserving the $\log n$ levels and distributing uniformly the alternatives, that is, the number of alternatives is the same in each level, possibly excepting the higher level.

4.3 Step 3: RBF Approximation

A Radial Basis Function (RBF) network is an artificial neural network that uses radial basis functions as activation functions. The output of the network is a linear combination of radial basis functions of the inputs and neuron parameters. Given certain mild conditions on the shape of the activation function the RBF networks are universal approximators on a compact subset of $\mathbb{R}^n$ which means that a RBF network with enough hidden neurons can approximate any continuous function with arbitrary precision.

For training the RBF network $\hat{\mathcal{U}}$ which approximates the utility function $\mathcal{U}$ the alternatives within the domain $\mathcal{D}$ are employed as inputs and the ranking level of each alternative, as outputs. The shape of the function $\mathcal{U}$ is captured by the ranking procedure and the artificial neural network has the whole of constructing a function which approximates this shape and introduces answers to other alternatives within the same domain.

The NN-DM model is trained in a domain standardized by scaling each dimension between zero and one. This standardization is required to make the

---

[1] The function $\log x$ is employed in this paper as representing the function $\log_2 x$.

tuning parameters easier. Once the model is constructed it is adjusted to the domain $\mathcal{D}$.

All data processing has been performed off-line employing the commercial software package MATLAB© [6]. For the construction of the RBF network the *newrb* function has been chosen with parameters given by Table 6.

Table 6. MATLAB parameters of the *newrb* function to construct the RBF network

Name	Value	Name	Value
P	Set F	SPREAD	500
T	Partial ranking	MN	200
GOAL	0	DF	25

4.4 Step 4: Performance Assessment

The original NN-DM method relies on the Kendall-Tau Distance (KTD) as an efficiency metric. The KTD is a metric that counts the number of pairwise disagreements between two ranking lists. In the NN-DM method these lists are generated by sorting the available alternatives according to the DM's underlying utility function and the resulting NN-DM model. The resulting KTD value has the following property: the smaller the value, the better the result. In the context of the original NN-DM method the KTD is an applicable metric because an underlying utility function is available to provide information about the quality of the resulting model.

As this paper focus in a real DM there is no underlying utility function which demands another validating process. The advantage is that here the process is validated by the DM himself. Once the model for the DM's preferences is constructed it is applied to sort the available data and the DM can verify the results accuracy.

4.5 Algorithm

Algorithm 1 presents the adapted NN-DM method to the real scenario introduced in Section 3. A grid of alternatives is constructed in the domain $\mathcal{D}$ provided by the DM. The decision-making matrix $\mathcal{M}$ is filled by the answers provided by the DM related to the alternatives belonging to the grid. A total ranking $\mathcal{R}$ is constructed and then clustered into $\log n$ levels. The RBF network converts the ranking $\mathcal{R}$ into a function $\hat{\mathcal{U}}$ able to provide answers to alternatives belonging to the entire domain $\mathcal{D}$. The NN-DM model is now fit to be employed in the estimates of each Pareto-optimal front.

Algorithm 2 introduces the NN-DM model applied to the polymer extrusion process. In each considered scenario the PFE and the corresponding model $\hat{\mathcal{U}}$

Algorithm 1 Adapted NN-DM method

1: Read the domain $\mathcal{D}$
2: Read the decision-making matrix $\mathcal{M}$
3: Built the total ranking of alternatives $\mathcal{R}$
4: Classify the alternatives into $\log n$ levels
5: Construct the RBF network $\hat{\mathcal{U}}$
---

are loaded. The model $\hat{\mathcal{U}}$ is then employed in evaluating each solution generating a sorting of the solutions from the best to the worst one.


Algorithm 2 NN-DM model applied to the polymer extrusion process
---
1: Load the Pareto-optimal front estimates
2: Load the NN-DM model $\hat{\mathcal{U}}$
3: Evaluate each available solution
4: Sort the solutions from the best to the worst one

Figure 3a presents an illustrative example in which the DM is represented by an underlying utility function $\mathcal{U}$ expressed as a Gaussian. After the construction of the NN-DM model the DM is not required to provide any data related to his preferences within the domain $\mathcal{D}$. Once the the resulting model $\hat{\mathcal{U}}$ is estimated it can be employed in quantifying any alternative within its domain, as shown in Figure 3b. From this point forward the alternatives can be sorted from the best to the worst one according to the DM's preferences represented by the NN-DM model.
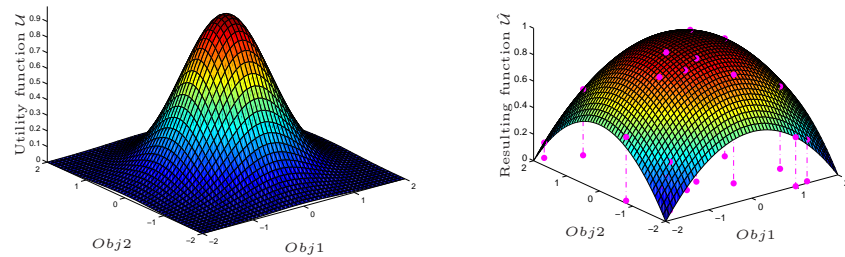

Fig. 3. (a) DM's underlying utility function $\mathcal{U}$ (b) Example of an application of the resulting function $\hat{\mathcal{U}}$

5 Computational Experiments

The filled decision-making matrices (Tables 4 and 5) provided by the DM are taken into account to construct general NN-DM models as described in Section 4. Figures 4 and 5 present respectively the models for the two considered scenarios: mass output × power consumption and mass output × WATS. The models have been trained in the provided domain $\mathcal{D}$ (Table 2).
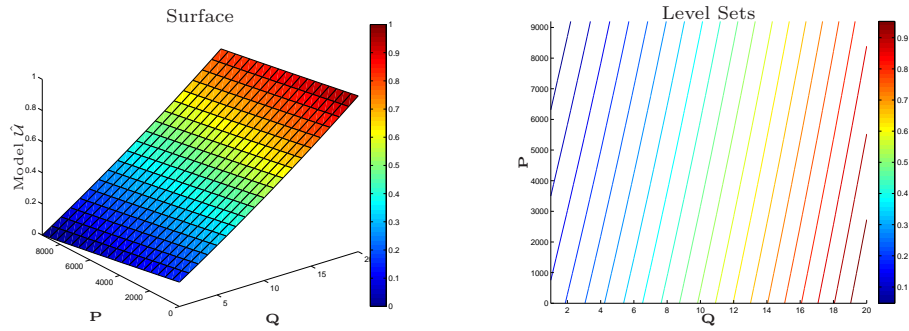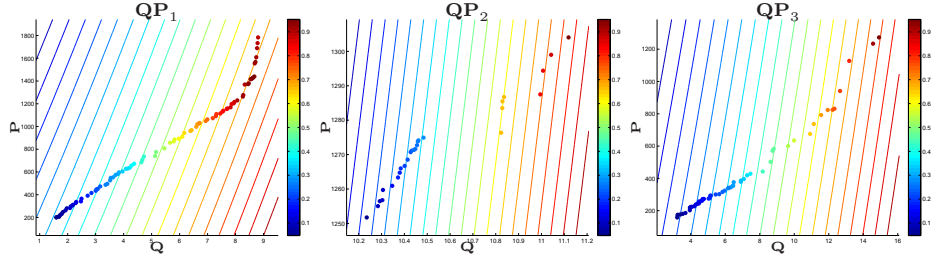

Fig. 4. General NN-DM model – mass output × power consumption
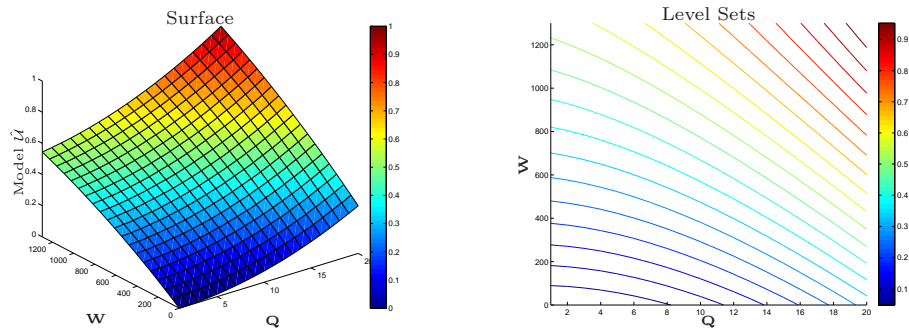

Fig. 5. General NN-DM model – mass output × WATS

Figures 6 and 7 present the general NN-DM models applied to sort the estimates of the Pareto-fronts considering the objectives mass output and power consumption ($\mathbf{QP}_1$, $\mathbf{QP}_2$, and $\mathbf{QP}_3$) and mass output and WATS ($\mathbf{QW}_1$, $\mathbf{QW}_2$, and $\mathbf{QW}_3$). The models' level sets are illustrated in the figures and the DM's preferences are represented by the external scale.

Fig. 6. NN-DM model applied to sort the estimates of the Pareto-optimal fronts in the problem mass output × power consumption
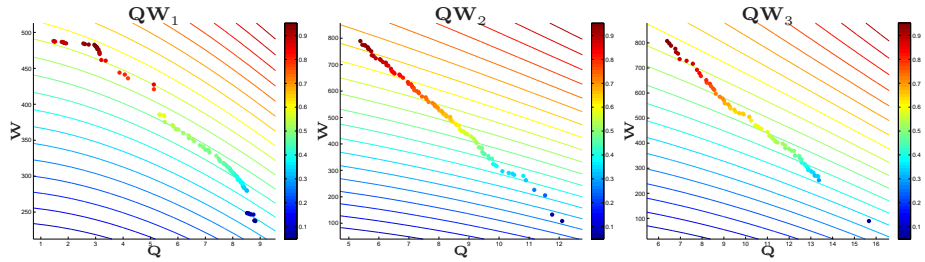
Fig. 7. NN-DM model applied to sort the estimates of the Pareto-optimal fronts in the problem mass output × WATS

The DM's preferences, captured by the decision-making matrices, are now represented by the NN-DM models which are employed in sorting the solutions belonging to the estimates of the Pareto-fronts. Moreover, the resulting models are now available to represent the DM's preferences in any other situations without demanding further information from the DM.

6 Conclusions

The NN-DM method is an algorithm which finds a model for the DM's preferences employing artificial neural networks. The NN-DM method assumes that there exists a utility function $\mathcal{U}$ which represents the DM's preferences in the provided domain $\mathcal{D}$ and constructs a model $\hat{\mathcal{U}}$ for these preferences by pairwise comparisons expressing ordinal relations only. Adaptations in the NN-DM method have been executed to obtain a model for the DM's preferences in the polymer extrusion process.

Six estimates of different Pareto-optimal fronts derived from the polymer extrusion process were examined in this paper. The results shown that the adapted NN-DM method is able to construct models that correspond to the DM's expectation in each considered scenario with a reasonable demand. Once the model

$\hat{\mathcal{U}}$ is trained it can be employed in quantifying any alternatives according to the DM's preferences and sort them from the best to the worst one. Furthermore, the resulting models $\hat{\mathcal{U}}$ can replace the DM in recurrent decisions with alternatives within the trained domain $\mathcal{D}$.

The authors are studying improvements in the NN-DM method to consider a more complete polymer extrusion process. The average melt temperature of the polymer at die exit ($\mathbf{T_{melt}}$) and the length of screw required to melt the polymer ($\mathbf{L_{melt}}$) also characterize the process performance and could have been considered in the multi-objective optimization problem. However, in a five-objective problem the decision-making matrix is inappropriate since it is difficult for a person to decide between two alternatives considering five conflicting objectives. Therefore, it is necessary a different approach to extract information from the DM. Additionally, since the optimization problem in this real scenario is computationally expensive, the NN-DM model may be employed in guiding the optimization process directly to the most preferable region avoiding computational effort expended in the non-preferable regions.

References

1. Deb, K.: Multi-objective optimization using evolutionary algorithms. John Wiley & Sons, Inc., New York, NY, USA (2001)
2. Deb, K., Sinha, A., Korhonen, P.J., Wallenius, J.: An interactive evolutionary multi-objective optimization method based on progressively approximated value functions. IEEE Transactions on Evolutionary Computation 14(5), 723–739 (2010)
3. Gaspar-Cunha, A.: Modelling and optimisation of single screw extrusion using multi-objective evolutionary algorithms. Lambert Academic Publishing (LAP), Koln, Germany (2009)
4. Gaspar-Cunha, A., Covas, J.A.: RPSGAe – Reduced Pareto Set Genetic Algorithm: application to polymer extrusion. In: Metaheuristics for Multiobjective Optimisation, pp. 221–249. Springer. Lecture Notes in Economics and Mathematical Systems Vol 535, Berlin Heidelberg (2004)
5. Köksalan, M., Karahan, I.: An interactive territory defining evolutionary algorithm: itdea. IEEE Transactions on Evolutionary Computation 14(5), 702–722 (2010)
6. MATLAB: Version 6.9.0 (R2009b). The MathWorks Inc., Natick, Massachusetts (2009)
7. Pedro, L.R., Takahashi, R.H.C.: Modeling decision-maker preferences through utility function level sets. In: 6th International Conference on Evolutionary Multi-criterion Optimization. Lecture Notes in Computer Science, vol. 6576, pp. 550–563. Springer Berlin Heidelberg, Ouro Preto, Brasil (2011)
8. Veldhuizen, D.A.V., Lamont, G.B.: Multiobjective evolutionary algorithms: analyzing the state-of-the-art. Evolutionary Computation 8(2), 125–147 (2000)