Applied Soft Computing 15 (2014) 42–56

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

An integrated approach to automated innovization for discovering useful design principles: Case studies from engineering

Kalyanmoy Deb[a,*], Sunith Bandaru[b], David Greiner[c], António Gaspar-Cunha[d], Cem Celal Tutum[a]

[a] Department of Electrical and Computer Engineering, Michigan State University, East Lansing, 428 South Shaw Lane, 2120 EB, MI 48824, USA
[b] Virtual Systems Research Centre, University of Skövde, SE 54128 Skövde, Sweden
[c] Institute of Intelligent Systems and Numerical Applications in Engineering (SIANI), Universidad de Las Palmas de Gran Canaria, 35017, Spain
[d] Institute for Polymer and Composites – I3N, University of Minho, Campus de Azurém, 4800-058 Guimarães, Portugal

ARTICLE INFO

Article history:
Received 15 February 2012
Received in revised form
19 September 2013
Accepted 8 October 2013
Available online 31 October 2013

Keywords:
Multi-objective optimization
Innovization
Design principles
Extrusion process
Noise barrier optimization
Friction stir welding

ABSTRACT

Computational optimization methods are most often used to find a single or multiple optimal or near-optimal solutions to the underlying optimization problem describing the problem at hand. In this paper, we elevate the use of optimization to a higher level in arriving at useful problem knowledge associated with the optimal or near-optimal solutions to a problem. In the proposed *innovization* process, first a set of trade-off optimal or near-optimal solutions are found using an evolutionary algorithm. Thereafter, the trade-off solutions are analyzed to decipher useful relationships among problem entities automatically so as to provide a better understanding of the problem to a designer or a practitioner. We provide an integrated algorithm for the innovization process and demonstrate the usefulness of the procedure to three real-world engineering design problems. New and innovative design principles obtained in each case should clearly motivate engineers and practitioners for its further application to more complex problems and its further development as a more efficient data analysis procedure.

Published by Elsevier B.V.

1. Introduction

Quest for new knowledge about problems of interest to an engineer or scientist has always been of utmost importance. However, due to constraints on time and other resources, practitioners are most interested in arriving at a single solution that will suffice the requirements for the instant. In a routine problem solving scenario such as in a design or a process operation activity, practitioners often need to solve an identical problem repeatedly but for different parameter settings. In such activities, instead of repeatedly executing similar tasks (which can be somewhat monotonous to an intelligent mind), a more wise approach would be to gather useful knowledge and problem properties that constitute a high-performing solution. Such knowledge will go a long way in providing insights about the problem and making the person an expert in solving the problem under consideration. In this paper, we suggest and discuss a computational approach for arriving at such useful knowledge thorough the use of an optimization process.

The elicitation of knowledge can be different in different problems. Here, we are particularly interested in knowledge that may help a designer or a practitioner in understanding their problems better. Often such knowledge can be thumb-rules or other rules such as decision-trees or semantic nets involving a few decision variables and problem functionalities. The important constituent of our approach is that the knowledge being extracted must be true for, not any arbitrary solution set, but for high-performing solutions of the problem. High-performing solutions are the solutions that are optimal or near-optimal corresponding to one or more objectives of the problem. This is where the need for an optimization algorithm arises.

When an optimization problem is formed for a single objective function, usually there is a single optimal solution that most optimization applications attempt to find. What we propose here is a multi-objective optimization study in which at least two conflicting objectives are considered. For example, cost of fabricating a product and its quality are two usual conflicting objectives of design. The advantage of considering multiple conflicting objectives is that the resulting optimization problem gives rise to a set of trade-off Pareto-optimal solutions [1–3]. Each of these solutions is optimal (and hence high-performing) with respect to certain trade-off

[*] Corresponding author.
E-mail addresses: kdeb@egr.msu.edu (K. Deb), sunith.bandaru@his.se (S. Bandaru), dgreiner@iusiani.ulpgc.es (D. Greiner), agc@dep.uminho.pt (A. Gaspar-Cunha), tutum@egr.msu.edu (C.C. Tutum).

1568-4946/$ – see front matter. Published by Elsevier B.V.
http://dx.doi.org/10.1016/j.asoc.2013.10.011

among the objectives. Since all these solutions are optimal and high-performing, an analysis of them may reveal important properties that they share. Such properties then can be considered as knowledge that high-performing solutions possess. Often, such knowledge brings in new concepts and innovative ideas of solving the problem optimally. Due to these possibilities, the task of searching for multiple trade-off solutions and identifying properties commonly appearing to these solutions is called as an *innovization* process – creating innovation through optimization.

There are some related studies in the data-mining and machine learning literature. However, most of these studies only provide information that can be perceived visually. For example, self-organizing maps have been used to project the multi-dimensional objective and design spaces onto a two-dimensional map, followed by hierarchical clustering to reveal clusters of similar design solutions [4]. Taboda and Coit [5] used *k*-means clustering on the trade-off solutions to simplify the task of analyzing them. Dendograms are used to depict strongly related decision variables in [6]. MODE or multi-objective design exploration [7] uses a combination of kriging and self-organizing maps to visualize the structure of the decision variables using the non-dominated solutions. Heatmap visualization inspired from biological micro-array analysis was proposed [8]. For many-objective problems, Walker et al. [9] proposed 'Pareto shells' and analyzed various methods for ordering the solutions. Oyama et al. [10] used proper orthogonal decomposition to decompose the design vector into the mean and fluctuation vectors.

Other studies aim at representing knowledge in the form of 'if-then' type rules using association rule-mining [11] or rough-set theory [12]. Such information, though very helpful in specific cases, is not compact. A typical multi-dimensional dataset may give rise to many such rules since no clustering procedure is invoked to highlight only the most important set of rules. The use of decision trees for representing the knowledge contained in large datasets also suffers from similar drawbacks [12]. Methods based on functional analysis of variance (ANOVA) [13] are only useful for considering each feature of the dataset one at a time. Correlations between different features are difficult to identify using ANOVA. While neural networks are very effective in modeling non-linear correlations between multiple inputs, the black box nature of the obtained networks may not be attractive to a practitioner. In this study, our goal is to extract knowledge from the trade-off dataset of any given multi-objective optimization problem. More importantly, the extracted knowledge should be simple, compact and significant. The methodology should be capable of identifying inter-dependencies between different problem entities (variables and objective functions) of the dataset and representing them in the form of closed-form mathematical expressions, so that any practitioner can remember them as thumb-rules for creating a good design in future design scenarios having a similar underlying structure.

It is important here to differentiate our method from data-modeling techniques such as regression, multivariate adaptive regression splines (MARS) [14], response surfaces and kriging [15] which also condense data in the form of closed-form mathematical expressions. Since the main purpose of these methods is to model the given data as closely as possible, they are not designed to look for abrupt changes in correlations. Being quite flexible, they are capable of adapting the resultant mathematical function to fit the differently correlated part of the data. An instance of this effect in MARS has been studied in [16]. On the other hand, our methodology has been tailored to weed out those parts of the input dataset that are either outliers or show an abrupt change in relationship.

The remainder of this paper is organized as follows. We briefly describe the proposed innovization process in Section 2. In Section 3, we describe a clustering based optimization technique for knowledge extraction, termed as automated innovization. All computer algorithms required for the integrated innovization process are outlined in this section. Thereafter, we consider three different real-world engineering design problems and apply the proposed automated innovization process and reveal useful knowledge about each problem. In all cases, the extracted knowledge provided new concepts of design which were not known before.

2. *Innovization*: Innovation through optimization

Designers and practitioners are often interested in solving their current problem at hand in order to meet deadlines and pre-specified targets. However, by virtue of their scientific bend of mind, they are always interested in gathering useful knowledge about their problem. The type and extent of knowledge can be different in different problems, but practitioners interested in engineering design problems would most likely be interested in knowing what design principles must a solution have in order for it be an optimal or high-performing solution. Such questions are vitally important to a designer as the answers to such questions provide deep insights among parameter interactions that would elevate a design to become optimal.

In the past few years, the first author has proposed a two-step procedure for unveiling such important information about a problem. The first step involves finding a set of high-performing solutions and the second step involves analyzing the obtained solutions to reveal important design principles. We discuss each of these two steps in the following paragraphs.

1 *Finding a set of high-performing solutions:* A design task usually involves a number of design variables each of which needs to be determined in order for the design to be feasible to be used and to achieve a certain goal. The goal is often to minimize the cost of fabrication, weight of the product, operation time, amount of harmful gas etc. The feasibility of a design is often checked by investigating if the design satisfies a number of pre-defined constraints, such as maximum stress developed due to loading is smaller than or equal to the strength of the material used or natural frequency of vibration is set well above the applied forcing frequency. Clearly, achieving such a feasible and optimal solution is not possible by manual (or trial-and-error) setting of variables, rather a computer-aided optimization algorithm is called for. Because of vagaries of design variables, constraints and goal functions, it becomes important to design or customize a suitable optimization algorithm for a particular problem. However, if a single goal is considered in the optimization task, the outcome would be a single optimal solution (we refer here as a high-performing solution). In the context of discovering design principles, we would require not one, but multiple high-performing solutions. An important question then to ask is where from multiple high-performing solutions will come? One way to look at the problem is again to follow what designers usually do in practice.

A designer in practice usually solves a similar problem repeatedly but with different parameter values. Let us take a typical scenario of an engineer who works in a pressure vessel design company. Today, the engineer may need to design a pressure vessel for a goal of minimum volume and for a particular internal pressure requirement for a refinery, tomorrow the same engineer may be designing another vessel for different internal pressure requirement for another petrochemical industry, and so on. By multiple solutions, we mean the optimal solution for each such scenario that the engineer is faced with every now and then in his/her work. One way to find multiple such solutions would be to treat the problem as a bi-objective optimization problem in which in addition to volume being a goal, we can include maximizing internal pressure as another conflicting goal. Theoretically, such a bi-objective consideration should find multiple trade-off solutions, each of which is an optimal solution.

2 *Analyzing solutions to reveal relationships:* After a set of trade-off solutions are found, the next step is usually taken to choose a single preferred solution by using subjective considerations and multi-criterion decision making principles [2,17–22]. However, here we suggest analyzing these solutions to unveil common principles hidden in them. Such principles, being common to high-performing trade-off solution, will dictate properties that would ensure Pareto-optimality and hence will indicate valuable properties related to the problem.

In the past few years, the first author has applied the above two steps of innovization to a number of engineering and other problem solving tasks [23–33]. In these studies, the first task was achieved using an evolutionary multi-objective optimization (EMO) technique and the second task was achieved by using manual regression fit of trade-off data. Certainly, such manual tasks are time consuming and are capable of bringing out relationships between two and at most three entities. An automated data mining procedure for the second task provides us with a hope of finding relationships involving multiple entities. However, the task is not trivial and far more complex than usual regression tasks for a number of different reasons:

1 First, we are interested in finding a mathematical relationship between variables, objective and constraint values of trade-off solutions obtained by an EMO and no information about the structure of such relationships are known *a priori*.

2 Second and importantly, most EMO-obtained datasets are likely to be close to the true Pareto-optimal set and may not be on the Pareto-optimal set. Thus, the methodologies for deriving relationships from near-optimal dataset are expected to remove noisy data and must be more challenging than the deterministic regression analysis methods which use the whole data.

3 Third, a relationship may exist parametrically to different subsets of the entire dataset. Thus, instead of one fixed relationship valid for the entire dataset, there can be multiple parametrically varying relationships that are valid to different subsets with different parameter values.

4 Fourth, the algorithms are expected to find multiple relationships so as to have a clear picture of how all the entities are interrelated.

5 Fifth, there are possibilities of lower and higher-level principles [34] that can be attempted to be learned from a given dataset.

3. Automated innovization

In the following sections, we describe how each task listed above is handled by an automated innovization algorithm, an early version of which has been suggested in the recent past by the authors [35,16].

3.1. Mathematical structure of relationships

Automated innovization extends the mathematical structure of relationships obtained by manual innovization to higher dimensions using the following representation,

$$\prod_{j=1}^{N} \phi_j(\mathbf{x})^{a_j b_j} = c. \tag{1}$$

Here, ϕ_j's are the symbolic entities (variables, objectives, constraints or any other functions of $\mathbf{x}$) which can combine in various ways to form parametric relationships common to all or most of the Pareto-optimal front. With N such 'basis functions' any combination can be generated by the Boolean variables a_j. Additionally, these functions can have exponents b_j's. The reasons for using such a mathematical structure are as follows: firstly, we have empirical evidence [23,24,26,27,29,30,32] that such a form is more common than others as far as design principles are concerned. Secondly, this form greatly simplifies the task. Thirdly, the form in Eq. (1) is scale-invariant which allows us to reduce the search space using appropriate transformations as shown in the next section. Most importantly, Eq. (1) falls into the category of 'power laws'. There is evidence [36] that many natural, physical, biological and man made processes follow power laws, easily identified by a straight line on logarithmic plots. Here, we have simply extended power laws to higher dimensions and hope to find all relationships that fit into this structure. Other problem specific mathematical structures may also be used. However, in the absence of such information, the above form should be used due to its resemblance to power laws.

3.2. Handling multi-modality

The expression on the left hand side in Eq. (1) can be called a commonality principle, only if it gives the *same* value for all Pareto-optimal vectors $\mathbf{x}^*$ or utmost changes only between subsets while remaining constant within them. Thus, c can be treated as a parameter, whose value is not as important as it remaining constant on the whole or subsets of the front.

Now, let us consider the variables b_j's. For generic problems, they can take any real value $(-\infty, \infty)$. However, since we are not concerned with the actual values of c, any set of b_j's that are in proportion to the set which actually forms a commonality principle, will also exhibit the features of commonality. To obtain relationships in a standard form and to restrict the search space of b so as to remove such multi-modalities, we suggest the following transformations,

$$b_j = \frac{b_j}{\{b_p \mid p = (arg\,max_p|a_p b_p|)\}}, \tag{2}$$

to have $b_j \in [-1, 1]\ \forall\, j = 1$ to N. Note that the c values also transform during this process in accordance with Eq. (1).

3.3. Dealing with near-Pareto optimality

Most multi-objective problem solving techniques are numerical algorithms and therefore the obtained trade-off fronts are never truly Pareto-optimal. This means that the c values evaluated above can never be *exactly* same. However, a low spread of c values definitely indicates that the evaluated expression is a valid principle. The lower the spread, the better the accuracy of that relationship. It is clear that this calls for an optimization task with the spread of c values as the objective and a_j's and b_j's as variables.

3.3.1. Clustering to identify subsets

As discussed before, c values can change parametrically over the Pareto-optimal front. Clustering of these c values is a viable method for identifying such subsets. Most clustering techniques require pre-specification of number of clusters [37,38]. It has been referred to as the "the fundamental problem of cluster validity" [39]. Although literature suggests different methods for determining the optimal number of clusters, in this paper we are able to circumvent this problem by integrating a special type of clustering called *grid-based clustering* [37] with the search process for design principles as will be shown later in Section 3.4.2. Grid-based clustering works by superimposing a grid on the input data points. For n-dimensional data, an n dimensional grid is required. Number of grid elements in each dimension are specified by n different parameters. Here, the c values have only one dimension. Let,

d: be the number of grids (or divisions as called in this paper),
m: be the number of Pareto-optimal points supplied, and
P_t: be the number of points in the t-th division.

The clustering procedure is as follows:

Step 1: Sort all m c-values obtained by evaluating Eq. (1) for all m trade-off solutions in the dataset.
Step 2: Divide the range $[c_{min}, c_{max}]$ into d divisions.
Step 3: Count the number of points in each division.
Step 4: Label those divisions that satisfy the following criterion as *sub-clusters:*

$$\text{Clustering criterion}:\ P_t \geq \left\lfloor \frac{m}{d} \right\rfloor = \text{Average points per division.} \tag{3}$$

Label all points within any division that does not satisfy Eq. (3) as *unclustered points*.
Step 5: Merge adjacent (in the dimension of c values) sub-clusters to form clusters.
Step 6: Count the number of clusters $\mathcal{C}$ and the total number of unclustered points $\mathcal{U}$.

Note that $1 \leq d \leq m$ is a requirement for correct clustering for avoiding zero-element clusters and zero number of clusters. Once $\mathcal{C}$ clusters are identified, the spread of c values should be calculated in each cluster separately.

3.4. Optimization for revealing multiple principles

The mixed variable nature of the optimization proposed above and the clustering procedure involved in the calculation of spread, limits the use of conventional optimization methods. The Boolean nature of a_j's suggests the use of binary strings in genetic algorithms (GAs).

3.4.1. Niching

The population based nature of GA also enables us to obtain *multiple* principles in the final population. Without niching only the best principle will survive, but by introducing the *niched-tournament* selection operator [40], selection tournaments can be restricted to solutions of the same kind (i.e. involving the same basis functions), thus promoting solutions with different sets of basis functions to coexist in the population. This can be implemented by comparing the binary strings of the population members involved.

3.4.2. Optimization problem formulation

The c values are scaled for each population member due to the scaling of b_j's. Since the objective function depends on the spread of these c values, it is important to ensure that all population members have a normalized measure of spread. In this work we use the percentage coefficient of variance $c_v = (\sigma/\mu) \times 100\%$ as this normalized measure. This value is calculated in all the $\mathcal{C}$ clusters and summed. For best clustering results, it is also necessary that clusters which are close yet disjoint (for example, due to gaps in the Pareto-optimal front) be combined. To achieve this we include the number of clusters in the objective function. By considering the parameter d as an additional variable in GA, we allow a very flexible clustering approach and at the same time alleviate the user from choosing this parameter.

A binary + real GA that employs (i) the new niched-tournament selection operator for handling the constraints, (ii) one-point crossover and bit-wise mutation for the binary string of a_j bits, (iii) simulated binary crossover (SBX) [41] and polynomial mutation [1] for b_j's and, (iv) a discrete version of SBX and polynomial mutation for the variable d, is used to solve the following automated innovization problem for extracting multiple principles simultaneously,

$$\begin{aligned}
\text{Minimize}\quad & \mathcal{C} + \sum_{k=1}^{\mathcal{C}} c_v^{(k)} \times 100\%, \qquad c_v^{(k)} = \frac{\sigma_c}{\mu_c} \quad \forall\ c \in k\text{-th cluster} \\
\text{Subject to}\quad & -1.0 \leq b_j \leq 1.0 \quad \forall\ j : a_j = 1, \\
& 1 \leq d \leq m, \\
& |b_j| \geq 0.1 \quad \forall\ j : a_j = 1, \\
& 1 \leq \sum_j a_j \leq \mathcal{N}, \\
& \mathcal{U} = 0, \\
& S = \frac{m - \mathcal{U}}{m} \times 100\% \geq S_{reqd}, \quad \mathcal{U}\ \text{recalculated using}\ P_t \geq \left\lfloor \frac{m}{d} \right\rfloor + \epsilon, \\
& a_j's\ \text{are Boolean},\quad b_j's\ \text{are real and}\ d\ \text{is an integer.}
\end{aligned} \tag{4}$$

The objective function is a weighted combination of the number of clusters (a positive integer) and the sum of percentage coefficient of variation in these clusters (which lies in the range 0–100). The use of percentage values helps to bring both terms of the objective function to the same order of magnitude. This allows an approximately unbiased weighting. A multi-objective formulation is avoided in favor of using niching to obtain multiple design principles in a single optimization run.

The first and second set of constraints are due to the imposed bounds. The third and fourth set of constraints are necessary to discourage trivial relationships. When all the exponents are close to zero, a case of low variance c values is obtained. This problem can be mitigated by restricting the exponents to a certain value. For the engineering examples considered in this paper and previous works [16,35,34], we have experimented with this limiting value and found that a minimum absolute exponent of 0.1 yields non-trivial relationships. Hence we have fixed this value to be 0.1 in this study. Recently however, the authors proposed an altogether different mathematical structure for the relationships [42] that completely eliminates the need to specify such a value. Another trivial case is a relationship with no basis functions, leading to a flat distribution of c values. This can be avoided by imposing a lower bound on $\sum_j a_j$. An upper bound $\mathcal{N}$ can also be used to specify the maximum number of basis functions that a relationship can contain. The fifth constraint $\mathcal{U} = 0$, ensures that d is sufficiently high so that even low-element (1 or 2) clusters are possible. The advantage of having a finer grid is that the resulting relationships are more accurate since the variance within clusters will be small. However, too large a value will cause number of clusters to increase in general. A population member that is close to an actual design principle would converge to it so as to balance the two terms in the objective. The last constraint enables a user to set a minimum requirement (S_{reqd}) for the significance S of the commonality principles. For calculating S, clustering is again performed but with a stricter criterion, so that low-element clusters can be identified.

3.5. Integrated algorithm for innovization

In this section, we present the proposed integrated algorithm for automated innovization:

1 Algorithm 1 shows the basic structure of the multi-objective optimizer NSGA-II [43]. The final population matrix Q_{NGEN} is directly read by Algorithm 2 to carry out the innovization procedure in an integrated way. The algorithms for `nondominated-sort()` and `crowding-distance()` can be found in [43].
2 Algorithm 2 shows the proposed automated innovization approach in a GA framework. The output population matrix from

NSGA-II is fed to this algorithm along with additional user defined parameters. For the basis functions we use the variables and objectives. The other routines are:

`create()`: It uses the variable vectors $\mathbf{x}$ in Q_{NGEN} and evaluates all the basis functions (N) for all the m trade-off solutions in Q_{NGEN} to give a data matrix $D_{m\times N}$.
`initialize()`: Randomly initializes the GA population for automated innovization such that $a_{ij} = \{0, 1\}$ and $b_{ij} \in [-1, 1]$.
`grid-cluster()`: Perform grid-based clustering. See Algorithm 3.
`evaluate-obj-constr()`: Evaluate the objective and all constraints in Eq. (4) except the significance constraint.
`niched-tournament()`: Perform niched tournament selection operation.
`crossover()`: Perform SBX crossover on b_j and single point crossover on a_j.
`mutation()`: Perform polynomial mutation on b_j and bitwise mutation on a_j.

3 Algorithm 3 takes the sorted c values in C and uses the clustering criterion $\lfloor (m/d) \rfloor + \epsilon$ to calculate number of clusters $\mathcal{C}$ and the unclustered points $\mathcal{U}$. It also returns the cluster matrix (CM) which stores which points belong to which clusters.

- `associate()`: It associates the current set of P_t points to the $\mathcal{C}$-th cluster and updates CM.

Algorithm 1. Elitist non-dominated sorting genetic algorithm (NSGA-II).

```
Input: $P_0$, $POP$, $NGEN$
Output: $Q_{NGEN}$
set $t \leftarrow 0$, $Q_0 \leftarrow \Phi$
while $t < NGEN$ do
  $R_t = P_t \cup Q_t$
  $\mathcal{F}$ = nondominated-sort($R_t$)
  $P_{t+1} = \Phi$ and $i \leftarrow 1$
  while $|P_{t+1}| + |\mathcal{F}_i| \le POP$ do
    crowding-distance($\mathcal{F}_i$)
    $P_{t+1} = P_{t+1} \cup \mathcal{F}_i$
    $i \leftarrow i+1$
  end while
  Sort($\mathcal{F}_i$, $\prec_c$)
  $P_{t+1} = P_{t+1} \cup \mathcal{F}_i[1 : (POP - |P_{t+1}|)]$
  $Q_{t+1}$ = select-cross-mutate($P_{t+1}$)
  $t \leftarrow t+1$
end while
```

Algorithm 2. Automated innovization.

```
Input: $Q_{NGEN}$, $m$, $N$, $maxgen$, $popsize$, $\phi_j\ \forall j$, $\mathcal{N}$, $\epsilon$, $S_{reqd}$
Output: $DP_{maxgen}$
1:  $D_{m\times N}$ = create($Q_{NGEN}$, $\phi_1, \phi_2, \ldots, \phi_N$)
2:  $DP_1$ = initialize($a_{ij}$, $b_{ij}\ \forall j = 1 : N$ and $d_i$) $\forall i = 1 : popsize$
3:  $gen \leftarrow 1$
4:  while $gen < maxgen$ do
5:    for $i$ : =1 to $popsize$ do
6:      $b_{ij} \leftarrow \frac{b_{ij}}{\{b_{ip}\ \mid\ p=(\arg\max_p |a_{ip} b_{ip}|)\}}\ \forall j = 1 : N$
7:      $c_i^k = \prod_{j=1}^{N} D_{ij}^{a_{ij} b_{ij}}\ \forall k = 1 : m$
8:      $C$ = Sort($c_i^1, c_i^2, \ldots, c_i^m$)
9:      $\mathcal{C}_i$, $\mathcal{U}_i$, $CM_i$ = grid-cluster($C$,$m$,$d_i$,0)
10:     $DP_{i,gen}$ = evaluate-obj-constr($DP_{i,gen}$,$\mathcal{C}_i$, $\mathcal{U}_i$,$CM_i$)
11:     $\mathcal{C}_i$, $\mathcal{U}_i$, $CM_i$ = grid-cluster($C$,$m$,$d_i$,$\epsilon$)
12:     Evaluate significance constraint: $S_i = \frac{(m-\mathcal{U}_i)}{m} \times 100\% \ge S_{reqd}$
13:   end for
14:   $DP_{gen+1}$ = niched-tournament($DP_{gen}$)
15:   $DP_{gen+1}$ = crossover($DP_{gen+1}$)
16:   $DP_{gen+1}$ = mutation($DP_{gen+1}$)
17:   $gen \leftarrow gen+1$
18: end while
```

Algorithm 3. `grid-cluster`(C,m,d_i,ϵ)

```
Input: $C$, $m$, $d$, $\epsilon$
Output: $\mathcal{C}$, $\mathcal{U}$, $CM$
set $\mathcal{C} \leftarrow 0$, $\mathcal{U} \leftarrow 0$, $flag \leftarrow 1$, $CM \leftarrow \Phi$
for $t$ : =1 to $d$ do
  if $P_t < \lfloor (m/d) \rfloor + \epsilon$ then
    $\mathcal{U} \leftarrow \mathcal{U} + P_t$
    $flag \leftarrow 1$
  else
    $CM$ = associate($CM$,$\mathcal{C}$,$C$,$P_t$)
    if $flag$ = 1 then
      $\mathcal{C} \leftarrow \mathcal{C} + 1$
    end if
    $flag \leftarrow 0$
  end if
end for
```

Being an integrated approach that first solves the original design problem (using Algorithm 1) and uses its output to solve the automated innovization problem (using Algorithm 2), it would be beneficial to use the same optimization sub-routines (e.g. crossover, mutation, etc.) for both cases. For the three problems studied in this paper, the best results were obtained using GAs (as supported by the references given in each case study). Therefore the same GA sub-routines are also used for automated innovization.

3.6. *Computational complexity of Algorithm 2*

It is well-known that the computational complexity of Algorithm 1 (NSGA-II) is $O(M \cdot POP^2)$ [43] where M is the number of objectives and POP is the population size. In this section we calculate the complexity of automated innovization shown in Algorithm 2 in terms of m the size of the dataset, N the number of basis functions used and d the number of divisions.

Each generation of Algorithm 2 performs three major operations on all *popsize* population members:

1 *Calculation of c values (Step 7):* The evaluation of each row in the data matrix $D_{m\times N}$ requires N exponentiations and $(2N - 1)$ multiplications. Thus each c value requires $(3N - 1)$ basic operations. This operation altogether would require $m(3N - 1)$ basic operations. Thus the complexity is $O(mN)$.
2 *Sorting of c values (Step 8):* Algorithm 2 performs merge sort [44] on the m c-values calculated in the above operation. It has a worst case complexity of $O(m \log m)$ [44].
3 *Grid-based clustering (Step 9):* For d divisions, Algorithm 3 shows that the `associate()` routine can be called at most d times (when all points are clustered). This routine in turn uses at most m comparisons (when all points lie within the same cluster). Thus the complexity for grid-based clustering is $O(dm)$. Note that the maximum possible value for d is m, which gives the worst case complexity for clustering over all generations as $O(m^2)$.

Of the three operations, sorting is the most efficient and hence it does not govern the complexity. The overall worst case complexity of Algorithm 2 is therefore either $O(mN)$ or $O(m^2)$. In general, we would expect $m > N$, thereby giving $O(m^2)$ as the theoretical upper limit.

In the following sections, we use the above integrated innovization algorithm to solve three real-world design optimization problems. The obtained principles are evaluated by domain experts and important insights are drawn. Table 1 shows the parameters used for solving the three problems.

4. Case study 1: Noise barrier design

Noise pollution is a problem of concern in the majority of developed countries, leading to the formalization of legal codes, laws,

Table 1
Summary of parameters used for Algorithms 1 and 2.

Parameter	Case study 1	Case study 2	Case study 3
Algorithm 1			
Population size (*POP*)	200	100	100
Generations (*NGEN*)	1000	50	30
Crossover type	Uniform	SBX	SBX
Crossover prob. (p_c)	1.0	0.8	0.9
Mutation type	Uniform	Polynomial	Polynomial
Mutation prob. (p_m)	0.015 and 0.03	0.05	0.1
Algorithm 2			
Dataset size (m)	Size of Q_{NGEN} obtained from Algorithm 1		
Basis functions (N)	No. of objectives + no. of variables		
Population size (*popsize*)	500	500	500
Generations (*maxgen*)	500	500	500
Max. basis func. ($\mathcal{N}$)	3	3	3
Clustering criterion (ϵ)	9	3	3
Min. significance (S_{reqd})	80%	80%	80%
Crossover prob. (p_c)	0.9	0.9	0.9
Mutation prob. (p_m)	0.05	0.05	0.05

or ordinances that establish certain limits on noise, particularly in urban areas. The problem of diminishing and controlling acoustic pollution can be addressed in several ways depending on the location of noise, i.e. at its source, during its propagation and at the receiver end. The placement of noise barriers between high traffic areas and residential ones is a common way of mitigating the environmental impact of noise. Among the methods for simulating open noise propagation around noise barriers, the use of the Boundary Element Method (BEM) has been established with the pioneering work of Seznec [45]. Later, in the nineties, some researchers developed the modeling of noise barriers with BEM and the evaluation of their 'insertion loss' [46–48]. Generalization to three-dimensional space was proposed in [49]. A further development in this active research field during the 2000s was achieved by Branco et al. [50] who use the BEM to study the effect of varying the shape of rigid acoustic barriers on the insertion loss. Recently BEM has been coupled with other methods. Hampel et al. [51] couples the BEM with the ray-tracing procedure; Tadeu et al. [52] introduces the Traction BEM (TBEM) and also couples BEM/TBEM with the method of fundamental solutions [53]. Some recent applications of traffic/highway noise barrier designs with BEM modeling are Oldham and Egan [54] and Grubesa et al. [55]. In the last lustrum, interest has been focused in obtaining optimum designs by coupling the BEM modeling with global automated optimization methods such as evolutionary computation. For example, Duhamel [56], Baulac et al. [57], etc. Other global meta-heuristic methods such as simulated annealing have also been tried [58]. Particularly, the optimum design of Y-shaped noise barriers has been considered in [59–62] by adjusting the insertion loss (IL) spectrum at different frequencies with respect to a reference curve to minimize the following fitness function,

$$\text{Fitness function}\,(\mathbf{x}) = \sum\nolimits_{i}^{NFreq} \left(IL_i(x) - IL_i^R\right)^2, \qquad (5)$$

where $\mathbf{x} = \{y_1, y_2, y_3, x_2\}$ are the barrier design variables shown in Fig. 1,

IL_i: insertion loss in the third octave band center frequency for the evaluated Y-barrier profile,
IL_i^R: insertion loss reference in the third octave band center frequency,
NFreq: Number of frequencies (= 5 here).

The reference IL^R spectrum could be obtained either from improving the IL spectrum of a previous existing noise barrier by a certain percentage (as done in [59,60,62], where 15% and 30% values are used) or from the IL spectrum of another existing barrier (as done in [61], where simple straight barriers with higher effective heights than the searched design are used). Multi-objective optimization of noise barriers was introduced in [60–62], using the above fitness function, combined with the effective height of the barrier [60] and also with the barrier length [61,62]. In this paper, the approach taken in [60] has been followed, performing a multi-objective optimization which combines: (a) the minimization of difference between the IL spectrum of the searched Y-noise barrier design and a reference IL spectrum, with (b) the simultaneous minimization of the barrier's effective height H_{eff}. The method searches for the barrier shape design which best fits IL^R for each effective height value. The five frequencies taken into account are: 63, 125, 250, 500 and 1000 Hz. A schematic of the two-dimensional configuration for scalar wave propagation problems in the frequency domain is shown in Fig. 1.

The required variables to construct the Y-shape noise barrier are the *y*-coordinates of points A, B and C, and the *x*-coordinate of point B, the complete search space taking a trapezoidal shape. Here $H1 = 10$ m, $H2 = 50$ m, the search space width = 1 m and maximum effective height $H_{eff} = 2.9891$ m.

Twelve independent runs of the NSGA-II were executed. Differences with [60] include a population size of 200 individuals, a stopping criterion set to 1000 generations and a smaller variable

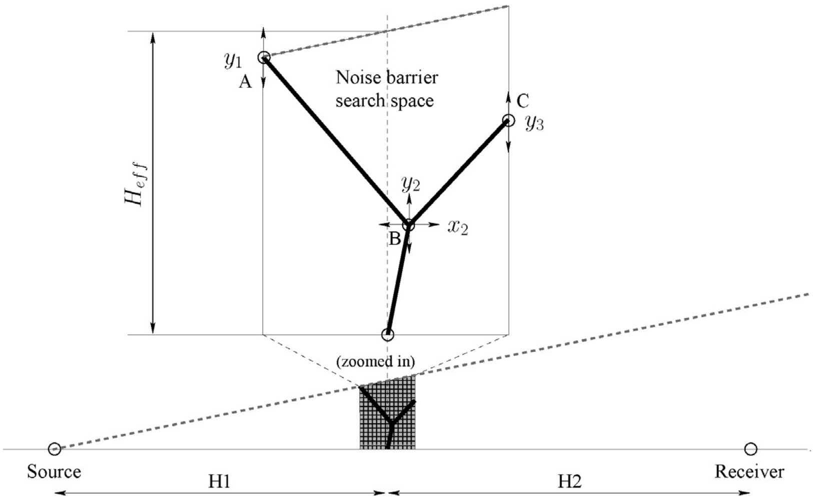

Fig. 1. Search space for the Y-shaped noise barrier design problem for case study 1.

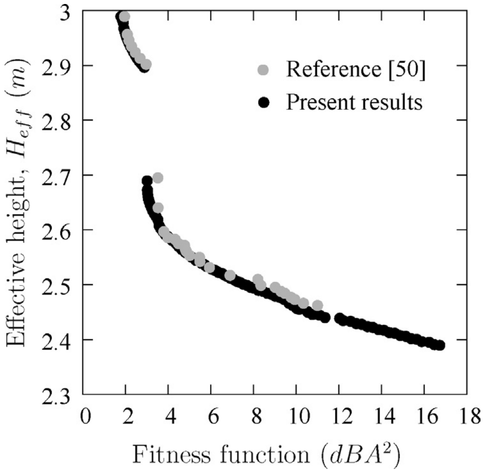

Fig. 2. Accumulated non-dominated solutions in objective space for case study 1.

range. Uniform crossover is used with gray coding and two different mutation rates of 1.5% (for first six runs) and 3% (for the other six runs). The non-dominated points accumulated from the final populations of the twelve runs are shown in Fig. 2 along with the reference results from [60]. The shapes reconstructed using BEM on these non-dominated noise barrier designs are shown in Fig. 3.

4.1. Results and discussion

From Fig. 3, some interesting visual information ('manual innovization') can be extracted that characterize the optimum solutions belonging to the non-dominated front. From the acoustical point of view, these automated designs seem to share a common shape: a very short trunk and two long arms which adapt to their maximum effective height, with the right arm being quasi-vertical. Considering that the IL reference spectrum here tends to force the noise barrier to its maximum noise attenuation capacity it can be said that, (a) it is beneficial to take advantage of the available effective height to its limit for both the arms and, (b) a quasi-vertical positioning of the right arm is advisable. When the automated innovization procedure is applied to this optimal noise barrier design problem, the following design principles are obtained from the non-dominated solutions, each being applicable to 85% of the optimum designs:

$$f_2^{-0.3361} x_2^{1.0000} = \text{constant} \tag{6}$$

$$y_1^{-0.3088} x_2^{1.0000} = \text{constant} \tag{7}$$

$$f_2^{-0.7441} x_2^{1.0000} y_3^{0.6921} = \text{constant} \tag{8}$$

$$y_1^{-0.6332} x_2^{1.0000} y_3^{0.5942} = \text{constant} \tag{9}$$

$$f_2^{-0.2235} y_1^{-0.1034} x_2^{1.0000} = \text{constant} \tag{10}$$

It is observed that the extracted design principles are all geometrical involving the second objective f_2. Relationships with the first objective – whose nature is acoustic related – are not found by this automated innovization procedure. Dividing Eq. (6) by Eq. (7), the following relationship between f_2 and y_1 can be achieved,

$$f_2 = (\text{constant}) \ \ y_1^{0.9188}. \tag{11}$$

When fitting a straight line to the whole dataset, the following equation is obtained:

$$f_2 = 0.9788 \ \ y_1 + 0.2105. \tag{12}$$

The relationship given by Eqs. (11) and (12) and the data are plotted in Fig. 4, where a complete fit can be observed (with constant in Eq. (11) equal to 1.146). Thus, a direct relationship between the y-coordinate of point A (see Fig. 1) and the effective height of each optimum design barrier is inferred. Therefore, the effective height is governed directly by the y-value of the left-most point: y_1. Geometrically speaking, this relationship makes sense, however the

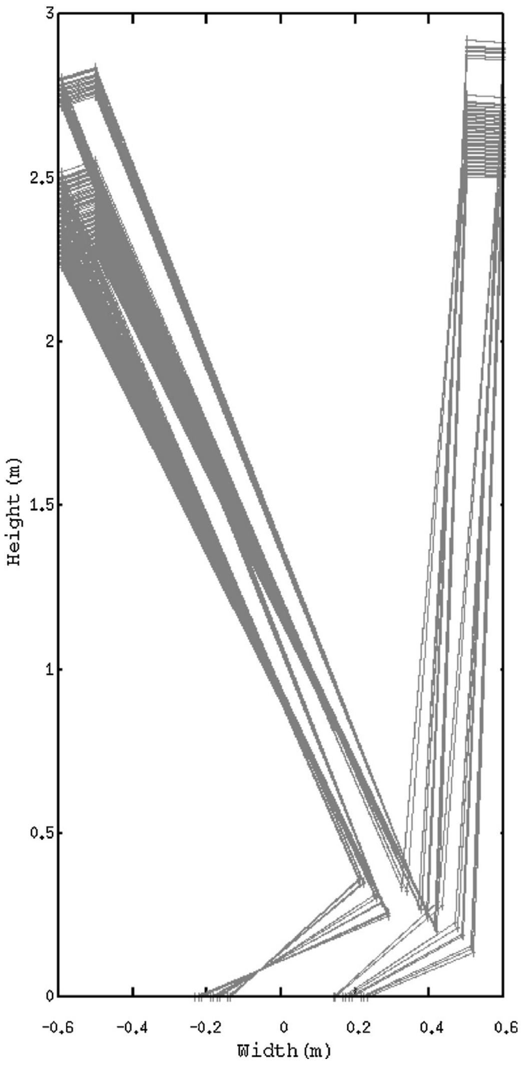

Fig. 3. Accumulated non-dominated noise barrier designs (variable space) for case study 1.

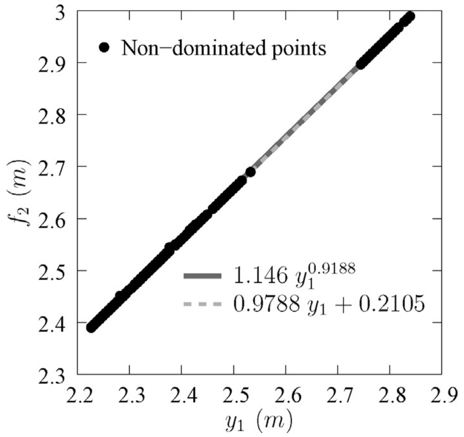

Fig. 4. Data points, automated innovization principle and the fitted straight line between y_1 and f_2 values for case study 1.

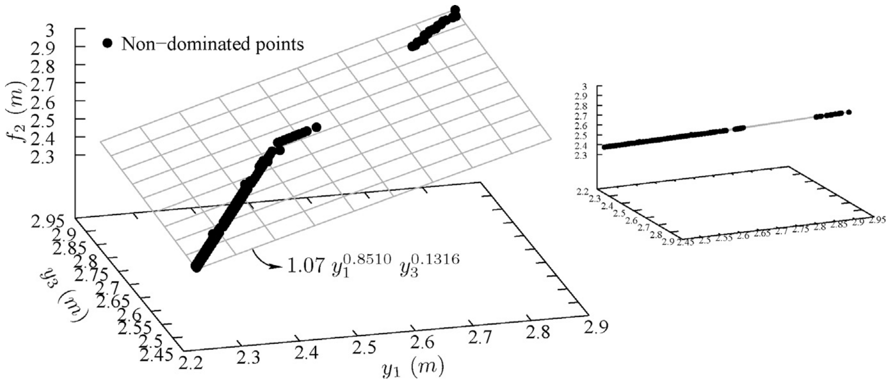

Fig. 5. Revealed relationship between y_1, y_3 and f_2 for case study 1. All non-dominated points lie on the surface specified by the relationship.

linear relationship between f_2 and y_1 obtained by our innovization study is a hallmark result.

In addition, when Eq. (8) is divided by Eq. (9) the following relationship between f_2, y_1 and y_3 is obtained:

$$f_2 = (\text{constant})\ \ y_1^{0.8510}\ \ y_3^{0.1316}. \tag{13}$$

The relationship given by Eq. (13) and the data plotted in Fig. 5 which again shows an adequate fit (with constant in Eq. (13) equal to 1.07). Note how the automated innovization approach could easily decipher a relationship in three dimensions where a manual innovization task would have been difficult. A direct correlation between y-coordinate of point A, the y-coordinate of point C (see Fig. 1) and the effective height of each optimum design barrier is thus inferred. From the automated innovization point of view, the procedure is flexible enough to be able to locate and reveal a close relationship among these variables.

Four out of five of the design principles obtained through automated innovization have been used to achieve the relationships shown in Figs. 4 and 5. They are in agreement with our first observation from 'manual innovization' in Fig. 3 that it is beneficial to take advantage of the available effective to its limit for both arms. The second observation that the right arm should be quasi-vertical turns out to be not so significant (given that $S_{reqd} = 80\%$) since according to the design principle in Eq. (6) x_2 slightly varies with the effective height. Nevertheless, the simplistic 'thumb-rule' type of relationships obtained by our study reveal valuable information for designing the noise barrier.

Fig. 6 shows the clusters obtained for design principle in Eq. (10) on the left and the mapping of these clusters on the trade-off front on the right. The two clusters are shown using different shades of gray. The unclustered points are denoted by '×'. It shows that the relationship is parametric with respect to the constant on the right hand. The cluster average c value for the clusters are 0.2768 and 0.3132.

5. Case study 2: Single screw polymer extrusion

Single screw extrusion is one of the most important polymer processing technologies, allowing the production of products such as pipes, films, profiles, and fibers. This process consists of feeding a solid polymer at the beginning of system (in the hopper), melting and homogenizing it and forcing the melted polymer to pass through a tool called the die that gives the final shape of the product to be obtained. Fig. 7 illustrates this process. The extruder constitutes of: (i) a hopper, where the solid polymer in the shape of pellets is fed, (ii) a heated barrel, (iii) an Archimedes-type screw rotating inside the barrel at a given speed (N), (iv) heater bands, whose temperature is defined by the operator and (v) a die [63,64].

The process performance depends on three different types of parameters: the polymer properties, the system geometry and the operating conditions. Different polymers are characterized by properties such as, thermal (e.g., heat conduction coefficient, melt temperature and heat capacity), physical (e.g., friction coefficients and density) and rheological (which is a measure of the resistance of the polymer to the flow). Usually, the process is optimized for a single polymer, thus the properties are, in this case, constant. In the most simple case, a conventional screw with three geometrical zones, as shown in Fig. 7, is used. First, a screw section with constant depth (H_1). Then, a compression section where the depth varies linearly between H_1 and H_3. Finally, a metering section with a constant depth but smaller (H_3). The screw is also characterized by the pitch (P) and the flight width (e). The operating conditions

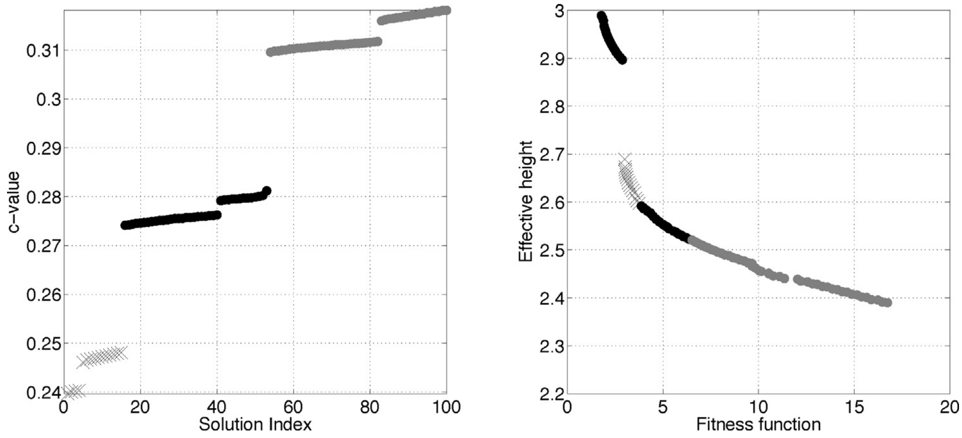

Fig. 6. Clusters obtained for design principle in (10) found using automated innovization.

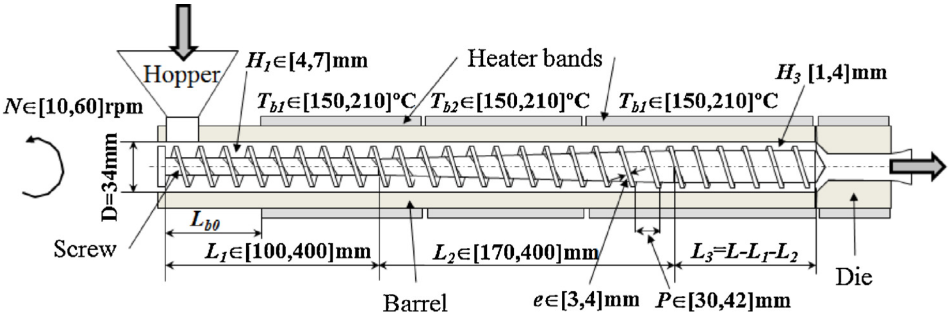

Fig. 7. Single screw extruder for case study 2.

are the variables that are controlled by the operator of the machine. In the present case these variables are the screw rotation speed (N) and the barrel temperature profile (T_{b1}, T_{b2} and T_{b3}) [63–65].

5.1. Process modeling

The modeling of this process must take into account that the polymer passes through different physical states,

1 *Solids conveying in the hopper:* First, the solids are fed in the hopper which by action of the gravity are transported inside the barrel.
2 *Solids conveying in the screw:* Then, 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.
3 *Delay zone:* At this point, due the heat generated by friction and the heat conducted from the barrel, a melt film is formed.
4 *Melting zone:* Then, a specific melting mechanism develops, characterized by the existence of a melt pool and melt films around the solid bed.
5 *Conveying zone:* Finally, the polymer is pressurized and is transported to the die.

The modeling of this process involves the linkage of all these functional zones using the appropriate boundary conditions. The program developed is based on the use of finite differences and is able to compute some important performance characteristics of the process as a function of the polymer properties, system geometry and operating conditions. The process performance is characterized by the mass output of the machine (Q), the average melt temperature of the polymer at die exit (T_{melt}), the mechanical power consumption required to rotate the screw (*Power*), the capacity of pressure generation (P_{max}), the mixing capacity measured by the average of deformation (*WATS*) and the length of screw required to melt the polymer (L_{melt}). More details of the modeling routine implemented can be found elsewhere [65].

5.2. Results and discussion

Two different type of studies were carried out as shown in Table 2. First, only the operating conditions (i.e., N, T_{b1}, T_{b2}, T_{b3}) were considered as variables for optimization (referred as Run 1). Then, the aim was to optimize the screw geometry (i.e., L_1, L_2, H_1, H_3, P, e) (referred as Run 2). Due to the complexity of the process and with the aim of better understanding the process concerning the optimization procedure, only two conflicting objectives were considered for each run. Table 3 lists the objectives used, the aim of optimization, and their range of variation (used as constraints).

Table 2
Optimization set-ups.

Run	Decision variables	Objectives
1	N, T_{b1}, T_{b2}, T_{b3}	Q, *Power*
2	L_1, L_2, H_1, H_3, P, e	Q, *Power*

Table 3
Objectives, aim of optimization, and range of variation for case study 2.

Objective	Aim of optimization	Range of variation
Output (kg/h)	Maximization	[1, 20]
Power consumption (W)	Minimization	[0, 9200]

Fig. 8 shows the non-dominated points obtained using NSGA-II. As expected, when the output increases the mechanical power consumption required to rotate the screw increases as well. In the case of Run 1, the extent of the Pareto-optimal front is wider because the screw speed is allowed to change and this is the decision variable that has the most important influence in the output. Table 4 shows the best results obtained after an application of the innovization algorithm described in this paper. In the case of Run 1, 82.83% of the points in the Pareto-optimal front are obtained when the value of barrel temperature T_{b2} is constant at about 210 °C. Fig. 9 shows that a single cluster containing 82.83% of the points is obtained for this design principle. Rest of the points remain unclustered. The mapping of these clusters and unclustered points on the trade-off front is shown on the right. Despite the fact that T_{b2} could have taken any value in [150, 210] °C, all Pareto-optimal solutions must take the highest value in order to be high-performing. This design principle provides a useful information to the designer and may provide new ideas of probably using a higher T_{b2} for even better operation of the extrusion process. Such information was difficult to comprehend before performing this study.

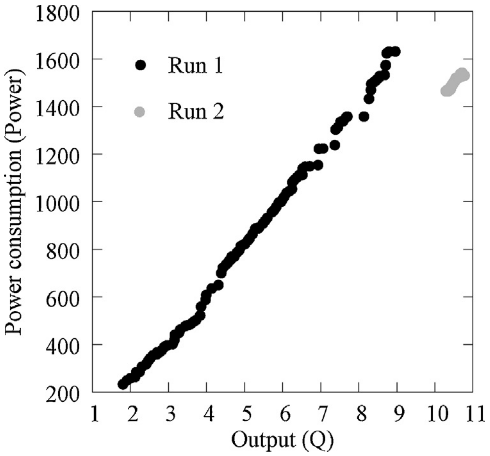

Fig. 8. Trade-off dataset obtained by NSGA-II for Runs 1 and 2 for case study 2.

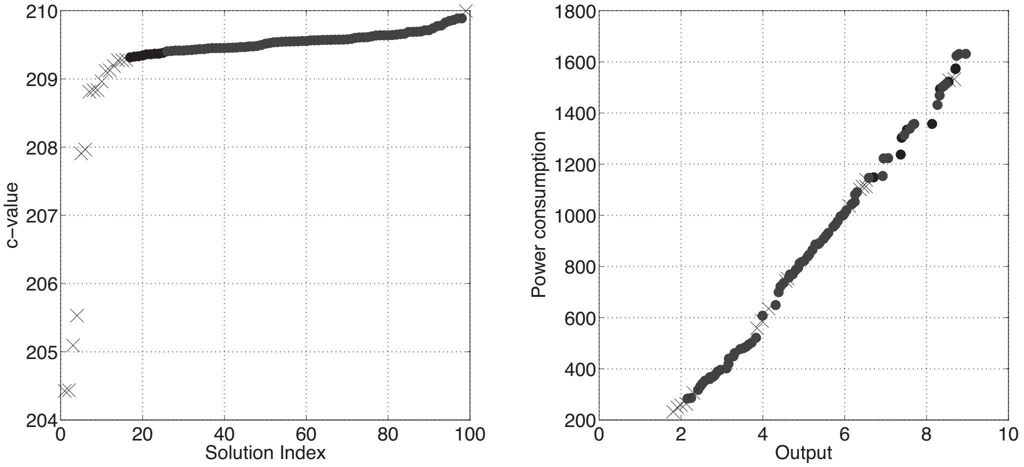

Fig. 9. Cluster obtained for design principle T_{b2} = constant. The cluster average c value is 209.5.

Table 4
Automated innovization results for case study 2.

Run	Relation	Results	Significance
1	(a)	$T_{b2}^{1.0000} = \text{constant}$	82.83%
2		With objectives	
	(a)	$L_2^{-0.6653} e^{0.1912} Q^{1.0000} = \text{constant}$	93.94%
	(b)	$L_2^{-0.5773} P^{-0.5486} Power^{1.0000} = \text{constant}$	93.94%
	(c)	$L_2^{-0.7943} Q^{-0.9251} Power^{1.0000} = \text{constant}$	93.94%
	(d)	$L_2^{1.0000} D_1^{0.1135} Power^{-0.9306} = \text{constant}$	93.94%
	(e)	$L_2^{1.0000} P^{-0.2794} Q^{-0.5416} = \text{constant}$	91.92%
	(f)	$L_2^{-0.5311} e^{1.0000} Power^{0.3688} = \text{constant}$	90.91%
	(g)	$L_1^{0.2363} L_2^{-0.6920} Q^{1.0000} = \text{constant}$	90.91%
	(h)	$L_1^{0.2979} L_2^{-0.9811} Power^{1.0000} = \text{constant}$	90.91%
	(i)	$L_2^{-0.6973} D_2^{-0.3618} Power^{1.0000} = \text{constant}$	90.91%
	(j)	$L_2^{1.0000} Q^{-0.5257} = \text{constant}$	90.91%
		Only variables	
	(k)	$L_1^{1.0000} L_2^{0.6194} e^{-0.7624} = \text{constant}$	93.94%
	(l)	$L_2^{1.0000} P^{-0.5010} = \text{constant}$	91.92%
	(m)	$L_2^{-0.5432} P^{0.2413} e^{1.0000} = \text{constant}$	90.91%
	(n)	$L_2^{-0.5622} D_1^{0.1830} e^{1.0000} = \text{constant}$	90.91%
	(o)	$L_2^{1.0000} D_1^{0.3127} P^{-0.5209} = \text{constant}$	90.91%

In the case of optimization of the screw geometry (Run 2), a considerable number of important design principles were obtained as shown in Table 4. Although a much smaller set of rules can be derived from the obtained set, here we indicate one or two important ones. Clearly, the design principles can be divided into two types, those which involve the objectives and those which involve only the decision variables. Most design principles (a, b, e, f, g, i and j shown in Table 4) indicate that Q and *Power* are approximately proportional to the $\sqrt{L_2}$. This is again a significant thumb-rule discovered by our study. Also, most revealed design principles involve the length of the compression zone (L_2). This is due to the fact that this length has a high importance in the polymer melting process, and as a consequence, in the performance concerning the two objectives considered here.

6. Case study 3: Friction stir welding

Friction stir welding (FSW) process is an efficient solid-state joining technique (i.e. without melting in the workpiece metal). The solid-state nature of the process avoids any problems related with cooling from the liquid phase such as porosity and solidification or liquation cracking, therefore making it very attractive especially for aluminum alloys which are difficult to weld with traditional welding techniques [66]. Fig. 10(a) shows a constantly rotating standard welding tool having a cylindrical shoulder with a probe which is submerged into butt joint between two clamped pieces and moved forward at a constant speed. The process can be simulated using different computational modeling approaches, with respect to different reference frames (see Fig. 10(b)), i.e. the Lagrangian, also known as the 'global approach' [67], where transient effects are captured, and the Eulerian, in other words the 'local approach' [68], in general used for the steady-state conditions.

In FSW, heat is generated by friction and plastic deformation. The amount of heat conducted into the work piece influences the quality of the weld, residual stress and deformation in the workpiece [67,68]. Insufficient heat generation from the tool shoulder and the probe could lead to failure of the tool pin during welding since the work piece material in front of the tool is not soft enough. Therefore, understanding the heat aspect of the FSW process is extremely important, not only for understanding physical phenomena, but also for improving the process efficiency, i.e. faster production in a robust framework.

6.1. Thermal FSW model

Due to relatively high heat generation contribution from the surface of the tool shoulder, it is assumed that the total heat generation is mainly composed of the heat produced only in the tool shoulder area [68]. In order to reduce the computational cost regarding moving heat source, meanwhile preserving the applicability, only the welding period is taken into account and a moving coordinate system (i.e. Eulerian reference frame) which is located on the heat source is applied. The shear layer formed below the tool shoulder due to high tool rotational speed and relatively high viscosity of the workpiece material is also included; hence an asymmetric temperature field along the joint line is obtained in the present numerical model (see Fig. 11(a)) by solving the heat conduction equation with proper boundary conditions assuming steady-state conditions.

The volumetric heat generation (q_{vol}) here is a function of the tool radius and the temperature dependent yield stress of the work piece material ($\sigma_{yield}(T)$ of AA2024-T3) as well as the tool rotational speed ($\omega = 2\pi n_{rev}/60$), and moreover assumed to be uniform through the thickness ($t = 3$ mm) of the plates to be welded. It is given by,

$$q_{vol} = \frac{\omega \sigma_{yield}(T) r(x, y)}{\sqrt{3} t}. \qquad (14)$$

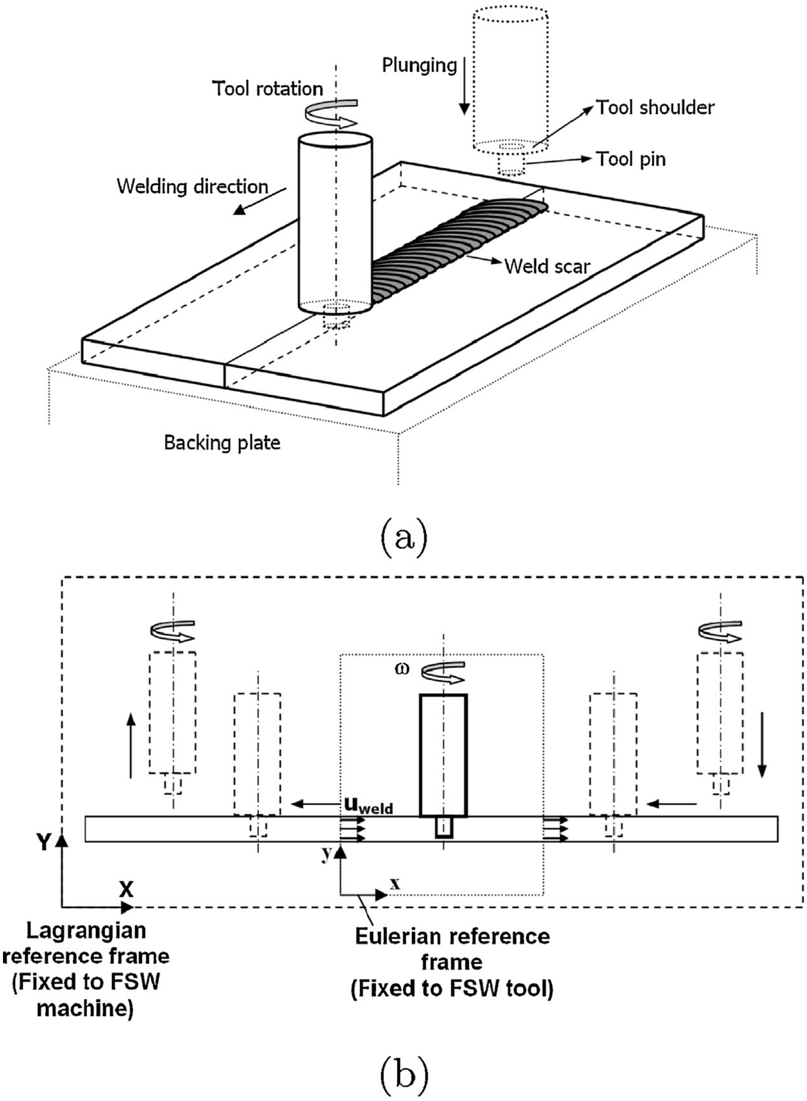

Fig. 10. (a) Schematic view of the FSW process for case study 3. (b) Lagrangian (global) approach versus Eulerian (local) approach for simulation of the FSW process.

The details of this temperature and position dependent heat source model entitled as Thermal-Pseudo-Mechanical (TPM) model are given in detail elsewhere [69]. The traverse motion of the tool as well as the relatively complex flow field under it are modeled by prescribing a material flow through the rectangular plate region, as shown in Fig. 11(a). The derivation of the mathematical prescription of the material flow is also schematically represented in Fig. 11(b) and components of the flow vector in the welding and the transverse directions are formulated for an arbitrary point on the periphery of tool shoulder as a function of θ

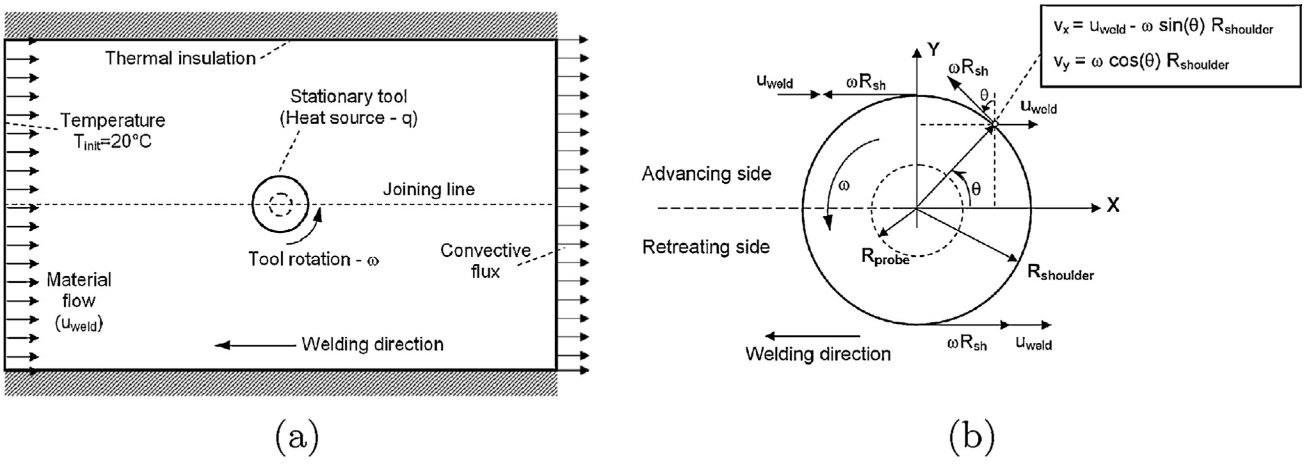

Fig. 11. (a) 2-D Eulerian steady state thermal model for case study 3. (b) Mathematical modeling of the flow field under the tool shoulder in detail.

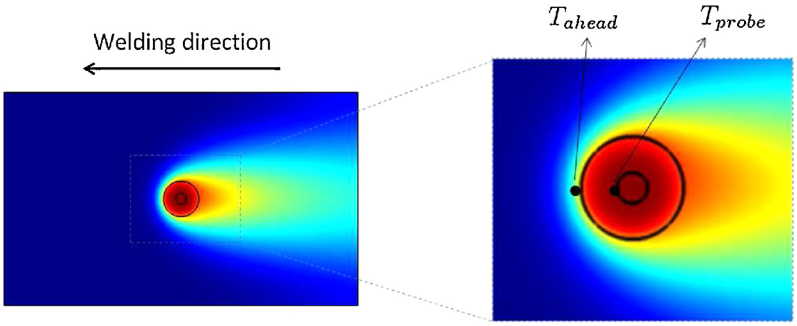

Fig. 12. Thermal field at steady-state conditions for case study 3.

(i.e. $\theta = \arccos(x/\sqrt{x^2+y^2})$ in Cartesian reference frame). Eq. (15) generalizes the flow field description ($u(\theta) = u(x, y) = (v_x, v_y)$) for the whole domain as follows,

$$u(x,y) = \begin{cases} (u_{weld} - sin(\theta)wr(x,y), cos(\theta)wr(x,y)), \text{ if } r(x,y) \leq R_{shoulder}; \\ (u_{weld}, 0), \text{ if } r(x,y) > R_{shoulder}, \end{cases} \tag{15}$$

where $r(x, y)$ is the radius or the position vector, $cos(\theta) = x/\sqrt{x^2+y^2}$ and $sin(\theta) = y/\sqrt{x^2+y^2}$. As a boundary condition, the room temperature (20 °C) is defined at the left edge of the rectangular region where the tool is assumed to be moving towards. The heat flux on the right edge of the plate region, where the material leaves the computational domain, is dominated by convection. On the upper and lower edges of the plate boundaries, thermal insulation is enforced. Fig. 12 shows an instance of an evolved thermal field (notice that T_{ahead} and T_{probe} are also mentioned to help understanding of the optimization problem which is formulated in Section 6.2).

6.2. Optimization problem

This section formulates the multi-objective optimization problem (MOP) briefly described above and related to the thermal aspects of the FSW process. Optimum process parameters, i.e. the tool rotational and traverse welding speeds (n_{rev} and u_{weld}), and geometrical tool parameters, i.e. tool shoulder and probe radii ($R_{shoulder}$ and R_{probe}), are investigated to maximize traverse welding speed, and simultaneously to minimize the temperature difference (ΔT) between the leading edge of the tool probe and the work piece material in front of the tool shoulder. The first objective is equivalently reformulated as the minimization of $-u_{weld}$. This MOP problem is constrained with hot and cold weld conditions, geometrical constraints (the tool shoulder radius is desired to be 5 mm larger than the tool probe radius), besides lower and upper limits of the design variables. In order to evaluate hot and cold weld conditions, average temperature (T_{avg}) is computed under the tool shoulder, in other words, the temperature values on each element inside the circular region (i.e. $Area = \pi(R^2_{shoulder} - R^2_{probe})$) are integrated and divided by the number of elements. The constrained multi-objective optimization problem is given below [68],

$$\begin{aligned} &Minimize: \; f_1(x) = -u_{weld}, \\ &Minimize: \; f_2(x) = \Delta T = T_{probe} - T_{ahead}, \\ &Subject\,to: \; 450\,^\circ\mathrm{C} \leq T_{avg} \leq 500\,^\circ\mathrm{C}, \\ &\qquad R_{probe} + 5\,\mathrm{mm} \leq R_{shoulder}, \\ &\qquad 8\,\mathrm{mm} \leq R_{shoulder} \leq 17\,\mathrm{mm}, \\ &\qquad 3\,\mathrm{mm} \leq R_{probe} \leq 12\,\mathrm{mm}, \\ &\qquad 100\,\mathrm{rpm} \leq n_{rev} \leq 1250\,\mathrm{rpm}, \\ &\qquad 0.5\,mm/s \leq u_{weld} \leq 15\,mm/s. \end{aligned} \tag{16}$$

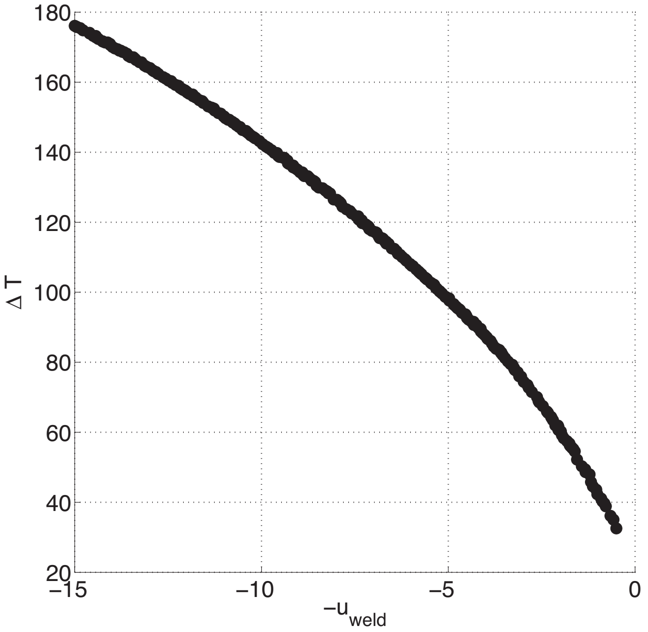

Fig. 13. Pareto-optimal front for the FSW problem (f_1: $Min. -u_{weld}$, f_2: $Min. \Delta T$) containing 287 non-dominated solutions for case study 3.

Table 5
Automated innovization results for case study 3.

Notation	Results	Significance
$DP1$	$R^{1.0000}_{shoulder} = \text{constant}$	95.12%
$DP2$	$u^{-0.5531}_{weld} \Delta T^{1.0000} T^{-0.1348}_{ahead} = \text{constant}$	94.08%
$DP3$	$u^{-0.5304}_{weld} \Delta T^{1.0000} T^{0.1392}_{probe} = \text{constant}$	93.38%
$DP4$	$u^{-0.5312}_{weld} \Delta T^{1.0000} = \text{constant}$	92.68%
$DP5$	$R^{1.0000}_{probe} = \text{constant}$	89.90%
$DP6$	$u^{-0.5303}_{weld} \Delta T^{1.0000} T^{0.1224}_{average} = \text{constant}$	88.50%
$DP7$	$u^{-0.9201}_{weld} n^{-0.5271}_{rev} T^{1.0000}_{average} = \text{constant}$	82.58%
$DP8$	$n^{-0.9831}_{rev} T^{1.0000}_{probe} T^{-0.7283}_{average} = \text{constant}$	81.88%
$DP9$	$u^{-0.5358}_{weld} n^{-0.8191}_{rev} \Delta T^{1.0000} = \text{constant}$	81.53%
$DP10$	$u^{-0.5304}_{weld} n^{-0.9775}_{rev} T^{1.0000}_{probe} = \text{constant}$	80.84%
$DP11$	$u^{-0.9833}_{weld} T^{1.0000}_{probe} T^{-0.4810}_{ahead} = \text{constant}$	80.14%
$DP12$	$u^{-0.9938}_{weld} T^{1.0000}_{probe} T^{-0.2496}_{average} = \text{constant}$	80.14%
$DP13$	$u^{-0.9938}_{weld} T^{1.0000}_{probe} = \text{constant}$	80.14%
$DP14$	$u^{-0.9967}_{weld} T^{1.0000}_{average} = \text{constant}$	80.14%

6.3. Results and discussion

Fig. 13 shows the Pareto-optimal front which is obtained by applying NSGA-II with a population size of 100 and run for 30 generations. Non-dominated solutions from different generations are archived to get 287 trade-off solutions.

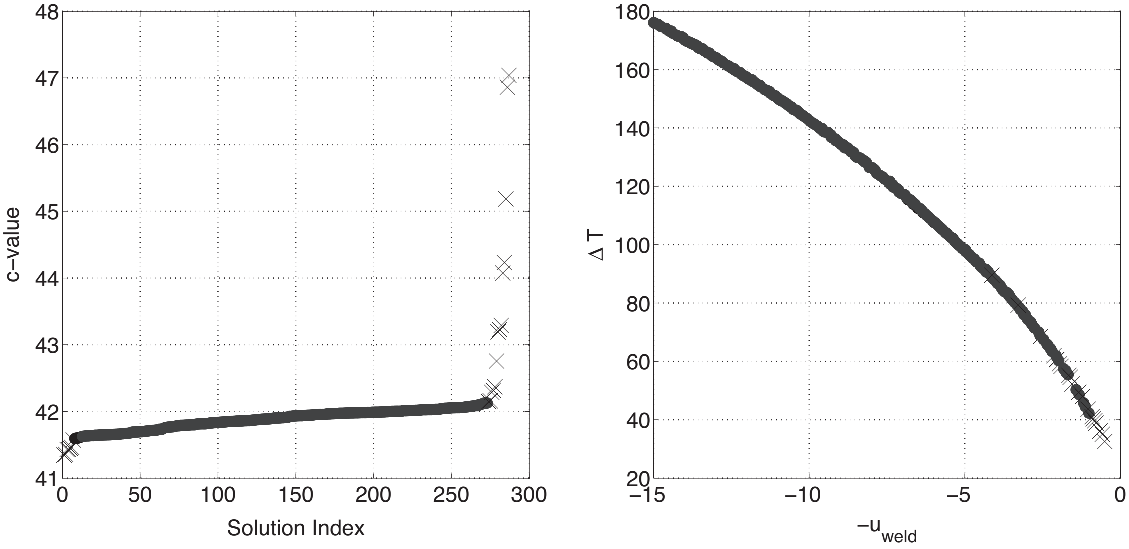

Fig. 14. Cluster plot for design principle *DP*4. The mapping shows that unclustered points occur for low weld speeds.

Table 5 shows the obtained design principles and their significance. A few striking aspects of the FSW problem that are revealed by these principles are:

1 *DP*1 and *DP*5 show that variables $R_{shoulder}$ and R_{probe} take the same values for about 95% and 90% of the trade-off solutions respectively and these values turn out to be the upper bounds of the variables (i.e., 17 mm and 12 mm, respectively).
2 Except for *DP*1, *DP*5 and *DP*8, u_{weld} appears in all the principles making it a very important and sensitive variable for the FSW process optimization. Inaccurate measurement or control of u_{weld} will thus effect all these relationships and the Pareto-optimal front may be disturbed significantly leading to sub-optimal operation. This crucial information is also supported by the previous studies by [67,70,71] in which u_{weld} has found out to be the most important process parameter in the FSW process affecting the transient thermal stresses and eventually evolution of the residual stresses in the final stage of the joint workpiece. This is because the traverse moving speed of the heat source (i.e. the FSW tool) directly governs the thermal gradients, therefore welding conditions (i.e. cold or hot weld conditions), together with the mechanical boundary conditions (e.g. clamping of the plate) and thermal softening behavior of the workpiece material, which in turn increase the thermal stresses, particularly in the longitudinal, i.e. welding, direction. In the present study, in a similar manner, calculation of the residual stresses have been bypassed with a pure thermal model by relevant problem formulation of minimizing the temperature difference (thermal gradients in front of the tool) between the tool and the workpiece.
3 n_{rev}, which is another important process parameter besides u_{weld}, appears in design principles *DP*7–*DP*10. However, these design principles involve more than two basis functions, which indicates the complexity or nonlinearity introduced in the heat generation function in Eq. (14) with the inclusion of this parameter. This can be explained as follows: An increase in n_{rev} initially results in higher heat generation thus increasing the temperature of the workpiece material. The temperature dependent yield stress of the workpiece material starts to counteract this through thermal softening, which highly complicates the prediction of the heat generation around the tool.
4 *DP*4 emphasizes the second design principle once more that an increase in u_{weld} results in an increase in ΔT [70] for almost 93% of the solutions. Similarly, slower welding speed (decrease in u_{weld}) causes the surrounding material to heat up more (due to the increase in time duration for the diffusion of the thermal signal) and consequently results in a lower ΔT. The cluster plot corresponding to *DP*4 is shown in Fig. 14. A single cluster containing 92.68% of trade-off solutions with an average *c* value of 41.88 is obtained. The right side plot shows that this relationship breaks at low weld speeds since the unclustered points occur to the right of the plot.
5 ΔT occurs in about 35% of the design principles, with an exponent of 1.0000 in all of them.
6 *DP*14 indicates that for most solutions, $T_{average}$ is (approximately) directly proportional to u_{weld}. The constant of proportionality is also revealed by the innovization study. Though the rise in $T_{average}$ with welding speed is expected, it is not intuitive that the approximated dependency is given by *DP*14.

All these principles are valuable for the practitioner involved with the welding process. Importantly, it is not clear how such important relationships can be obtained by any other means, other than finding a set of high-performing solutions and then analyzing them as portrayed in this paper.

7. Conclusions

Methods to gather useful knowledge about a problem always fascinated man. In engineering and scientific problem solving tasks, often users are engaged in finding a single solution for their problems at hand. In this paper, we have suggested a computational optimization based methodology for unveiling solution principles that are associated with most high-performing solutions of a problem. For this purpose, we have suggested a two-step procedure in which first a set of high-performing trade-off solutions are obtained using a multi-objective optimization technique and then these solutions are analyzed to reveal useful mathematical relationships among decision variables, constraint and objective functions that would guarantee a solution to be near-optimal. Although the usefulness of the proposed *innovization* task executed through manual means has been demonstrated amply in many engineering and scientific problem solving tasks in the past by the first author and his collaborators, in this paper, we have discussed recently proposed automated innovization procedures that use sophisticated clustering based optimization techniques to extract hidden solution principles.

The main hallmark of this study is the suggestion of a combined optimization cum analysis procedure based on elitist non-dominated sorting genetic algorithm (or NSGA-II) and a clustering based niched evolutionary algorithm. The procedure is applied to three real-world engineering design problems. In all three cases, new and innovative information about the problems have been discovered. These information are simplistic and 'thumb-rule' like and importantly none of these information was available before. Based on these results, we suggest further and immediate application of the proposed integrated innovization procedure and its extensions (such as lower-level innovization, higher-level innovization [34] and temporal evolution [72]) to other and more complex engineering design problems.

Acknowledgements

The first two authors acknowledge the financial support provided by Academy of Finland Grant 133387. The second author also acknowledges the support provided by Department of Electrical and Computer Engineering at Michigan State University.

The third author acknowledges Orlando Maeso and Juan J. Aznarez (SIANI, Universidad de Las Palmas de GC) for providing the acoustic modeling BEM software used to perform the calculations. Also the financial support provided in part by the research project UNLP08-3E.2010 of Secretaria de Estado de Universidades e Investigacion, Ministerio de Ciencia e Innovacin (Spain) and FEDER is greatly appreciated.

References

[1] K. Deb, Multi-objective Optimization Using Evolutionary Algorithms, Wiley, New York, 2001.
[2] K. Miettinen, Nonlinear Multiobjective Optimization, Kluwer, Boston, 1999.
[3] R. Steuer, Multiple Criteria Optimization: Theory, Computation, and Application, Krieger, Malabar, FL, 1989.
[4] S. Obayashi, D. Sasaki, Visualization and data mining of Pareto solutions using self-organizing map, in: Evolutionary Multi-criterion Optimization, Springer, 2003, pp. 796–809.
[5] H. Taboada, D. Coit, Data mining techniques to facilitate the analysis of the Pareto-optimal set for multiple objective problems, in: Proceedings of the 2006 Industrial Engineering Research Conference (CD-ROM), 2006.
[6] T. Ulrich, D. Brockhoff, E. Zitzler, Pattern identification in Pareto-set approximations, in: GECCO '08 – Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation, ACM, 2008, pp. 737–744.
[7] S. Obayashi, S. Jeong, K. Chiba, Multi-objective design exploration for aerodynamic configurations, AIAA Paper 4666, 2005.
[8] A. Pryke, S. Mostaghim, A. Nazemi, Heatmap visualization of population based multi objective algorithms, in: Evolutionary Multi-Criterion Optimization, Springer, 2007, pp. 361–375.
[9] D. Walker, R. Everson, J. Fieldsend, Visualisation and ordering of many-objective populations, in: 2010 IEEE Congress on Evolutionary Computation (IEEE-CEC), 2010, pp. 1–8.
[10] A. Oyama, T. Nonomura, K. Fujii, Data mining of Pareto-optimal transonic airfoil shapes using proper orthogonal decomposition, AIAA Paper 4000, 2009.
[11] K. Sugimura, S. Obayashi, S. Jeong, Multi-objective design exploration of a centrifugal impeller accompanied with a vaned diffuser, in: Proceedings of the 5th Joint ASME/JSME Fluid Engineering Conference, FEDSM2007-37502, 2007.
[12] K. Sugimura, S. Obayashi, S. Jeong, Multi-objective optimization and design rule mining for an aerodynamically efficient and stable centrifugal impeller with a vaned diffuser, Engineering Optimization 42 (3) (2010) 271–293.
[13] S. Jeong, M. Murayama, K. Yamamoto, Efficient optimization design method using kriging model, Journal of Aircraft 42 (2) (2005) 413–420.
[14] J.H. Friedman, Multivariate adaptive regression splines, The Annals of Statistics 19 (1) (1991) 1–67.
[15] T. Simpson, J. Poplinski, P. Koch, J. Allen, Metamodels for computer-based engineering design: survey and recommendations, Engineering with Computers 17 (2) (2001) 129–150.
[16] S. Bandaru, K. Deb, Towards automating the discovery of certain innovative design principles through a clustering-based optimization technique, Engineering Optimization 43 (9) (2011) 911–941.
[17] V. Chankong, Y.Y. Haimes, Multiobjective Decision Making Theory and Methodology, North-Holland, New York, 1983.
[18] S.M. Lee, Goal Programming for Decision Analysis, Auerbach Publishers, Philadelphia, 1972.
[19] C.-L. Hwang, A.S.M. Masud, Multiple Objective Decision Making – Methods and Applications: A State-of-the-art Survey, Springer-Verlag, Berlin, 1979.
[20] T.L. Saaty, Decision Making for Leaders: The Analytic Hierarchy Process for Decisions in a Complex World, RWS Publications, Pittsburgh, Pennsylvania, 2008.
[21] M. Ehrgott, B. Naujoks, T.J. Stewart, J. Wallenius, Multiple criteria decision making for sustainable energy and transportation systems, in: Proceedings of the 19th International Conference on Multiple Criteria Decision Making, Lecture Notes in Economics and Mathematical Systems, Springer, Berlin, 2010, p. 634.
[22] M. Zeleny, Multiple Criteria Decision Making, McGraw-Hill, New York, 1982.
[23] K. Deb, A. Srinivasan, Innovization: innovating design principles through optimization, in: GECCO '06 – Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation, ACM, New York, 2006, pp. 1629–1636.
[24] N. Padhye, K. Deb, Multi-objective optimisation and multi-criteria decision making in SLS using evolutionary approaches, Rapid Prototyping Journal 17 (6) (2011) 458–478.
[25] S. Chaudhuri, K. Deb, An interactive evolutionary multi-objective optimization and decision making procedure, Applied Soft Computing Journal 10 (2010) 496–511.
[26] K. Deb, K. Sindhya, Deciphering innovative principles for optimal electric brushless d.c. permanent magnet motor design, in: Proceedings of the World Congress on Computational Intelligence (WCCI-2008), IEEE Press, Piscatway, NY, 2008, pp. 2283–2290.
[27] K. Deb, S. Gupta, D. Daum, J. Branke, A. Mall, D. Padmanabhan, Reliability-based optimization using evolutionary algorithms, IEEE Transactions on Evolutionary Computation 13 (5) (2009) 1054–1074.
[28] S. Mittal, K. Deb, Optimal strategies of the iterated prisoner's dilemma problem for multiple conflicting objectives, IEEE Transactions on Evolutionary Computation 13 (3) (2009) 554–565.
[29] K. Deb, S. Tiwari, Multi-objective optimization of a leg mechanism using genetic algorithms, Engineering Optimization 37 (4) (2005) 325–350.
[30] K. Deb, K. Mitra, R. Dewri, S. Majumdar, Towards a better understanding of the epoxy polymerization process using multi-objective evolutionary computation, Chemical Engineering Science 59 (20) (2004) 4261–4277.
[31] K. Deb, P. Jain, N. Gupta, H. Maji, Multi-objective placement of electronic components using evolutionary algorithms, IEEE Transactions on Components and Packaging Technologies 27 (3) (2004) 480–492.
[32] K. Deb, S. Jain, Multi-speed gearbox design using multi-objective evolutionary algorithms, ASME Transactions on Mechanical Design 125 (3) (2003) 609–619.
[33] K. Deb, A.R. Reddy, S. Chaudhuri, A reliable classification of gene clusters for cancer samples using a hybrid multi-objective evolutionary procedure, in: S. Bandyopadhyay, U. Maulik, J.T. Wang (Eds.), Analysis of Biological Data: A Soft-computing Approach, World Scientific, New Jersey, 2007, pp. 232–257.
[34] S. Bandaru, K. Deb, Higher and lower-level knowledge discovery from pareto-optimal sets, Journal of Global Optimization (2013) 1–18, http://dx.doi.org/10.1007/s10898-012-0026-x.
[35] S. Bandaru, K. Deb, Automated innovization for simultaneous discovery of multiple rules in bi-objective problems, in: Proceedings of the 6th International Conference on Evolutionary Multi-criterion Optimization, EMO 2011, Springer-Verlag, Berlin, Heidelberg, 2011, pp. 1–15.
[36] M. Newman, Power laws pareto distributions and zipf's law, Contemporary Physics 46 (5) (2005) 323–351.
[37] P. Berkhin, Grouping Multidimensional Data. Chapter: A Survey of Clustering Data Mining Techniques, Springer-Verlag, Berlin, Heidelberg, 2006, pp. 25–71.
[38] R. Xu, D. Wunsch, Survey of clustering algorithms, IEEE Transactions on Neural Networks 16 (3) (2005) 645–678.
[39] C. Chen, Handbook of Pattern Recognition and Computer Vision, World Scientific, Singapore, 2010.
[40] C.K. Oei, D.E. Goldberg, S.J. Chang, Tournament Selection, Niching, and the Preservation of Diversity, IlliGAL Report No. 91011, University of Illinois at Urbana-Champaign, Urbana, IL, 1991.
[41] K. Deb, R. Agrawal, Simulated binary crossover for continuous search space, Complex Systems 9 (2) (1995) 115–148.
[42] S. Bandaru, K. Deb, A dimensionally-aware genetic programming architecture for automated innovization, in: Proceedings of the 7th International Conference on Evolutionary Multi-criterion Optimization, EMO 2013, Springer-Verlag, Berlin, Heidelberg, 2013, pp. 513–527.
[43] K. Deb, S. Agarwal, A. Pratap, T. Meyarivan, A fast and elitist multi-objective genetic algorithm: NSGA-II, IEEE Transactions on Evolutionary Computation 6 (2) (2002) 182–197.
[44] D. Knuth, The Art of Computer Programming, Addison-Wesley, 2006.
[45] R. Seznec, Diffraction of sound around barriers: use of the boundary elements technique, Journal of Sound and Vibration 73 (2) (1980) 195–209.
[46] D. Hothersall, S. Chandler-Wilde, M. Hajmirzae, Efficiency of single noise barriers, Journal of Sound and Vibration 146 (2) (1991) 303–322.
[47] J. Park, W. Eversman, A boundary element method for propagation over absorbing boundaries, Journal of Sound and Vibration 175 (2) (1994) 197–218.
[48] S. Chandler-Wilde, The boundary element method in outdoor noise propagation, Proceedings of the Institute of Acoustics 19 (1997) 27–50.
[49] D. Duhamel, Efficient calculation of the three-dimensional sound pressure field around a noise barrier, Journal of Sound and Vibration 197 (5) (1996) 547–571.
[50] F. Branco, L. Godinho, A. Tadeu, Acoustic insertion loss provided by rigid acoustic barriers of different shapes, Journal of Computational Acoustics 11 (4) (2003) 503–520.

[51] S. Hampel, S. Langer, A. Cisilino, Coupling boundary elements to a raytracing procedure, International Journal for Numerical Methods in Engineering 73 (3) (2008) 427–445.

[52] A. Tadeu, J. António, P. Mendes, L. Godinho, Sound pressure level attenuation provided by thin rigid screens coupled to tall buildings, Journal of Sound and Vibration 304 (3–5) (2007) 479–496.

[53] A. Tadeu, J. António, I. Castro, Coupling the bem/tbem and the mfs for the numerical simulation of acoustic wave propagation, Engineering Analysis with Boundary Elements 34 (4) (2010) 405–416.

[54] D. Oldham, C. Egan, A parametric investigation of the performance of t-profiled highway noise barriers and the identification of a potential predictive approach, Applied Acoustics 72 (11) (2011) 803–813.

[55] S. Grubesa, H. Domitrovic, K. Jambrosic, Performance of traffic noise barriers with varying cross-section, Promet-Traffic & Transportation 23 (3) (2011).

[56] D. Duhamel, Shape optimization of noise barriers using genetic algorithms, Journal of Sound and Vibration 297 (1/2) (2006) 432–443.

[57] M. Baulac, J. Defrance, P. Jean, Optimisation with genetic algorithm of the acoustic performance of t-shaped noise barriers with a reactive top surface, Applied Acoustics 69 (4) (2008) 332–342.

[58] S. Mun, Y. Cho, Noise barrier optimization using a simulated annealing algorithm, Applied Acoustics 70 (8) (2009) 1094–1098.

[59] D. Greiner, J. Aznárez, O. Maeso, G. Winter, Shape design of noise barriers using evolutionary optimization and boundary elements, in: Proceedings of the Fifth International Conference on Engineering Computational Technology, Civil-Comp Press, 2006.

[60] O. Maeso, D. Greiner, J. Aznárez, G. Winter, Design of noise barriers with boundary elements and genetic algorithms, in: Advances in Boundary Element Techniques IX (BETEQ-2008), 2008, pp. 101–106.

[61] D. Greiner, B. Galván, J. Aznárez, O. Maeso, G. Winter, Robust design of noise attenuation barriers with evolutionary multiobjective algorithms and the boundary element method, in: Evolutionary Multi-Criterion Optimization, Springer, 2009, pp. 261–274.

[62] D. Greiner, J. Aznárez, O. Maeso, G. Winter, Single-and multi-objective shape design of y-noise barriers using evolutionary computation and boundary elements, Advances in Engineering Software 41 (2) (2010) 368–378.

[63] Z. Tadmor, C. Gogos, Principles of Polymer Processing, 2nd ed., John Wiley and Sons, Hoboken, NJ, USA, 2006.

[64] C. Rauwendaal, ModPolymer Extrusion, 2nd ed., Munich, Germany, Hanser Verlag, 2001.

[65] A. Gaspar-Cunha, Modelling and Optimisation of Single Screw Extrusion – Using Multi-Objective Evolutionary Algorithms, 1st ed., Lambert Academic Publishing, Köln, Germany, 2009.

[66] TWI, Friction stir welding at TWI, 1991 http://www.twi.co.uk/content/fswintro.html

[67] C. Tutum, J. Hattel, A multi-objective optimization application in friction stir welding: considering thermo-mechanical aspects, in: IEEE Congress on Evolutionary Computation (CEC) 2010, IEEE Press, Barcelona, Spain, 2010, pp. 427–434.

[68] C. Tutum, K. Deb, J. Hattel, Hybrid search for faster production and safer process conditions in friction stir welding, in: Proceedings of the 8th International Conference on Simulated Evolution and Learning, SEAL'10, Springer-Verlag, 2010, pp. 603–612.

[69] H. Schmidt, J. Hattel, Thermal modelling of friction stir welding, Scripta Materialia 58 (2008) 332–337.

[70] C. Tutum, J. Hattel, Optimisation of process parameters in friction stir welding based on residual stress analysis: a feasibility study, Science and Technology of Welding & Joining 15 (5) (2010) 369–377.

[71] C. Tutum, J. Hattel, Multi-objective Evolutionary Optimisation for Product Design and Manufacturing. Chapter: State-of-the-art Multi-objective Optimisation of Manufacturing Processes based on Thermo-mechanical Simulations, Springer-Verlag, 2011, pp. 71–133.

[72] K. Deb, S. Bandaru, C. Celal Tutum, Temporal evolution of design principles in engineering systems: analogies with human evolution, in: Proceedings of the 12th International Conference on Parallel Problem Solving from Nature, PPSN 2012, Springer-Verlag, Berlin, Heidelberg, 2012, pp. 1–10.