A Soft Computing Approach to Mastering Paper Machines

Christer Carlsson
IAMSR, Åbo Akademi University
Joukahainengatan 3-5A
Turku, Finland
christer.carlsson@abo.fi

Matteo Brunelli
Systems Analysis Laboratory
Aalto University, P.O. box 11100
Aalto, Finland
matteo.brunelli@aalto.fi

József Mezei
IAMSR, Åbo Akademi University
Joukahainengatan 3-5A
Turku, Finland
jmezei@abo.fi

Abstract

Paper machines are extremely complex systems which require a deep knowledge of the relations between levels of usage of factors and characteristics of the process/final product. In this paper we propose to capture the tacit knowledge of experts in the form of a fuzzy ontology. Based on the fuzzy ontology, we introduce a knowledge-based system which can ultimately recommend what factors should be increased, or decreased, in order to obtain a different and better output. The system is based on an integer linear goal-programming optimization problem whose parameters come from the fuzzy ontology. We also propose extensions of our model to account for additional constraints and knowledge expressed in the form of fuzzy numbers.

1 Introduction

Modern industrial processes have become highly automated and increasingly complex as engineers try to squeeze the last percentages of productivity out of the machines. There are several reasons for that, but the main reason is that tougher competition is forcing down the prices of end-products at the same time as the costs for energy, raw material and components are increasing every year. This is eating away the profit margins step by step, which can be seen and foreseen and the answer to this is to try to improve productivity in order to retain a reasonable profit margin. Then there are external disturbances such as changes in the exchange rates that can take away significant parts of the margins in a very short time and crises of various types that can cut down on demand for shorter or longer periods and for which not much can be done to find compensations.

Increasingly complex industrial processes have been monitored and controlled by experienced engineers who are good at reacting to problems with good solutions in a short time, actually in sufficient time to avoid larger problems and expensive consequences; there has also been efforts to work out optimal solutions to problems in the production process and also to combine them with optimal timing of the problem solving processes. In the hunt for reduced costs many industrial corporations have now fired the experienced engineers and hired young, less costly but inexperienced engineers. The consequences became visible rather quickly and rather sharply: complex industrial processes meet with problems that are not properly solved (or not solved at all) which create large, complex and very costly new problems. In a large research program called Dyscotec with eight corporations representing the Finnish process industry we have worked on identifying these problems and have defined as a research objective to find models and methods which can translate the tacit knowledge of experienced engineers to advice and support functions for the new, young and inexperienced engineers that have been hired to run complex industrial processes.

In one of our cases we work with paper machines, which can be described as the set of interacting processes on Figure 1.

Figure 1: Paper machine - key processes [20]

The idea to translate the tacit knowledge of experienced engineers to advice and support functions for inexperienced process operators have been tried a number of times and (in the short version) without much success. The key problem appears to be the incompatibility of precision and relevance—when tacit knowledge is transformed to advice and support functions with sufficient precision for a true representation it becomes irrelevant for the users as they cannot master the complexity; on the other hand, if
it is made simple enough to guide the inexperienced operators it is not a true representation of the precise, tacit knowledge which exists on how to master complex processes.

One attempt to give guidance to the operators was built on colour codes for the key parts of the processes that were collected in Excel spreadsheets (see Figure 2).

In the rows are key parts of the production processes (first column) and factors that decide how the process parts work (second column); in the columns are process impacts of the factors; for each row we can follow the effects of increasing/decreasing a particular factor which are colour-coded: blue colour means that the factor is having a positive effect; orange colour means that the factor is having a negative effect; the strengths of the effects are shown with the shades of the colours (and also indicated with numbers). The idea is to train the operators on what combinations of factors will be the most productive for making sure that the paper machines runs effectively (which will be defined by multiple criteria).

1.1 Fuzzy ontology

Ontology can be defined as a systematic description of is-a relationships and entity dependencies. The information describing a paper machine (the processes and characteristics which are dependent on different process parts) can be structured in an ontological way. The information one can obtain concerning the relationships of factors and characteristics of a machine is usually imprecise and vague. In recent years, it has been widely pointed out that classical ontologies are not appropriate to deal with imprecise and vague knowledge, which is inherent to several real world domains [19].

To overcome the limitations of crisp ontologies, fuzzy ontologies started to appear in the beginning of 2000’s and proved to be efficient tools to handle many different real world problems: information retrieval [18], text summarization [13], traffic control [11], and knowledge mobilization [10, 20]. There exists numerous definitions of fuzzy ontology in the literature and usually these definitions are anchored to the specific application under consideration. A very general definition is the following: “a fuzzy ontology is simply an ontology which uses fuzzy logic to provide a natural representation of imprecise and vague knowledge and eases reasoning over it” [1]. In this paper, the fuzzy ontology is represented by means of a fuzzy relation as in [2].

1.2 Goal programming

Goal programming is a technique which was originally proposed to solve multi-criteria decision making problems [5, 6]. With the help of goal programming, we can handle multiple conflicting objectives within a linear programming formulation. In the optimization process, aspiration levels (target values) are defined for every objective which express the specific requirements of the decision maker. In the goal programming formulation of a multiobjective problem, the objective is to minimize the deviations from the given aspiration levels.

Goal programming was first used in fuzzy environment in [14]: fuzzy weights are associated with imprecisely specified goals. In recent years, fuzzy goal programming became a widely used approach within the multi-criteria decision-making. It has been applied to solve problems in different areas: waste management [4], water quality management [12], portfolio selection [17], traffic management [15], and water resource allocation [16].

In the following we will work out some steps in how to build an efficient advice and support system using soft computing methods. The paper is structured as follows: sections 2 and 3 give an overview of the context and the structure of the ontology, section 4 introduces the optimization model constructed to deal with the proposed problem, section 5 gives a short overview of possible extensions. In section 6, we provide a numerical example using Excel software. Finally, section 7 gives a short summary and conclusions.

2 Paper machines as systems

A paper machine is an extremely complex system, but can ultimately be described in terms of the characteristics of the outcome and the intensity of the factors used in the production process. In practice, there are a finite set of factors $F = \{f_1, \ldots, f_n\}$ and a set of characteristics of a process/products $C = \{c_1, \ldots, c_m\}$. Some examples of factors are the amount of steam, or any other substance, used in the making of the paper. Conversely, characteristics of the process/product could be the quality of the paper, its brightness but also the total cost of the process. We know that a change in the characteristics of the process can be obtained only by changing the actual setting of the factors as clearly, if the management leave the factors unchanged, then they should not expect a different output with respect to its characteristics.

A paper machine is, to some extent, a grey box where a lot of factors are used as inputs and a lots of characteristics are used to describe the output. We can indeed deduce that the characteristics of the process are functions of the factors used, since they depend uniquely on them. Figure 3 provides a schematic representation of a paper machine.
In our specific context we are given some data and we want to make sense of it. Such data represent the tacit knowledge of experts in paper machines. More formally, besides sets $C$ and $F$, the only available information is the following valued relation $\mu_R \colon F \times C \to L$,

\begin{equation}
\mu_R(f_i, c_j) = \begin{cases} 
1, & \text{if } f_i \text{ strongly positively affects } c_j \\
\alpha \in [0, 1], & \text{if } f_i \text{ to some extent positively affects } c_j \\
0, & \text{if } f_i \text{ does not affect } c_j \\
\beta \in [-1, 0], & \text{if } f_i \text{ to some extent negatively affects } c_j \\
-1, & \text{if } f_i \text{ strongly negatively affects } c_j 
\end{cases}
\end{equation}

where $L$ is an ordered set and $\mu_R(f_i, c_j) \in L$ describes the intensity of relation between a factor and a characteristic, represented on a bipolar scale [7]. A bipolar scale is nothing else but a scale with two polarities. A typical bipolar scale can be the scale measuring someone’s preferences on a pair of objects whereas a unipolar scale is suitable to measure phenomena such as temperature. In the case of the relation between factors and characteristics, the scale is naturally bipolar because a factor can either positively or negatively affect a characteristic. For instance, an increased amount of bleaching has a beneficial effect on the brightness of the paper, but a negative effect on the cost of the process. Using an interval, or even a scale with many possible values, is indeed an expedient to model the intrinsic imprecision of the process. To bring an example, it is reasonably that two factors, as bleaching and quantity of non-recycled paper used in the process, affect positively the quality of the final product, but too simplistic to assume that they do it with the same intensity. This is why one needs a scale to measure the degrees of relation between factors and characteristics.

In fact, the underlying reasoning is hardly ever binary since it is more appropriate that there exists degrees to which a factor affects the characteristic of a process. A possible solution in this direction would be that of assuming $L = [-1, 1]$ with the following associated semantic

However, to be even more consistent with reality, one must admit that, even for an expert, is very difficult to give the values $\mu_R(f_i, c_j) \in L$ with absolute certainty. In this case, in order to account for uncertainty, $L$ could for instance be assumed to be the set of fuzzy numbers with support $[-1, 1]$, or intervals, or probability distributions. To be precise, in our problem, the expert was asked to compile the ontology in the form of a cluster heat map (see [22] for a brief account). To make the ontology computable, the heat map was transformed into a numerical
one in the form of the relation (1). However, as it was very difficult to establish a certain correspondence between colors and values in $[-1, 1]$, we decide that using tools to incorporate uncertainty would have been natural.

Furthermore let us note that the relation associated with the membership function $\mu_R$ can be seen as an ontology. The most widely accepted definition of ontology is probably that given by Gruber [9] in the context of artificial intelligence: ‘An ontology is an explicit specification of a conceptualization’. Let us note that this definition is sufficiently ambiguous to give the opportunity to use a number of different tools to represent and handle it. We parenthetically note that relation $R$ does in fact satisfy the definition of ontology proposed in [2].

**Example 1.** Considering that a paper machine is a complex system of factors and characteristics, the following Table 1 could be an excerpt from a fuzzy ontology.

| Table 1: Example of relation $R$ | C1 | C2 | C3 | ...
|----------------------------------|----|----|----|-----
| F1                               | 1  | 0.5| 1  | ...
| F2                               | 0.1| -0.7| 0.6| ...
| F3                               | 0.5| 0.3| -0.6| ...
| ...                              | ...| ...| ...|     

*If we, instead, wanted to build a fuzzy ontology with triangular fuzzy numbers, then the result could look similar to the following Table 2.*

| Table 2: Example of relation $R$ with values of the membership functions under the form of triangular fuzzy numbers | C1       | C2       | C3       | ...
|---------------------------------------------------------------|----------|----------|----------|-----
| F1                                                            | (0.6,1,1)| (0.2,0.5,0.7)| (0.9,1,1)| ...
| F2                                                            | (-0.2,0.1,0.2)| (-0.8,-0.7,-0.3)| (0.5,0.6,0.9)| ...
| F3                                                            | (0.2,0.5,0.6)| (0.3,0.1)| (-0.8,-0.6,-0.3)| ...
| ...                                                            | ...      | ...      | ...      |     

3.1 Making sense of the data

After having been able to represent the knowledge of the expert, utilizing it becomes the real challenge. What we aim to do is to create a decision support system for other decision makers. To create such a system, we start from a simple consideration. Namely, given a set of characteristics $C = \{c_1, \ldots, c_m\}$, we can always *partition* it into

- $C_1$: the characteristics of the process that we want to increase
- $C_2$: the characteristics of the process that we want to decrease
- $C_3$: the characteristics of the process that we want to leave unaltered
- $C_4$: the characteristics of the process that we are indifferent to.

This partition always exists, also when the management is satisfied with the process and does not wish to alter the outcome. In fact, in this specific case we would have $C = C_3$. What we want to do is to associate a partition of the characteristics with a partition of factors. Given this partition, which approximately identifies a state that we want to reach, we want to find a suitable partition of factors which could tell us what factors one should enhance, decrease or leave unchanged in order to reach the desired outcome. Given a set of factors $F = \{f_1, \ldots, f_n\}$, we can partition it into three disjoint subsets

- $F_1$ factors to increase
- $F_2$ factors to decrease
- $F_3$ factors that we want to leave unaltered

which can give the decision maker some guidelines when he tries to reach the desired state. Figure 4 provides a snapshot.

![Figure 3: Paper machine](image1.png)

![Figure 4: A partition of the characteristics should induce a partition of factors](image2.png)
4 The Optimization Model

The task of determining the optimal partition of the factors can be formulated as a multiobjective optimization problem. The feasible region of the problem is the set of \( n \)-dimensional vectors, \( \mathbf{x} = (x_1, \ldots, x_n) \), such that \( x_j \in \{-1, 0, 1\}, j = 1, \ldots, n \). Every characteristic \( c_j \in C_1 \cup C_2 \cup C_3 \) corresponds to one objective function:

- if \( c_j \in C_1 \), then we want to maximize the function
  \[ g_j(\mathbf{x}) = \sum_{i=1}^{n} \mu_R(f_i, c_j)x_i \]

- if \( c_j \in C_2 \), then we want to maximize the function
  \[ h_j(\mathbf{x}) = -\sum_{i=1}^{n} \mu_R(f_i, c_j)x_i \]

- if \( c_j \in C_3 \), then we want to maximize the function
  \[ k_j(\mathbf{x}) = -\left|\sum_{i=1}^{n} \mu_R(f_i, c_j)x_i\right| \]

In order to find the optimal sets \( F_1, F_2, \) and \( F_3 \), one needs to find the (weakly) Pareto optimal solutions of the problem

\[
\max_{\mathbf{x} \in \{-1, 0, 1\}^n} F = (g_1, \ldots, g_{|C_1|}, h_1, \ldots, h_{|C_2|}, k_1, \ldots, k_{|C_3|}).
\]

Instead of solving the multiobjective optimization problem, we formulate the following single objective problem:

\[
\max_{\mathbf{x} \in \{-1, 0, 1\}^n} \sum_{j \in C_1} \lambda_j g_j(\mathbf{x}) + \sum_{j \in C_2} \lambda_j h_j(\mathbf{x}) + \sum_{j \in C_3} \lambda_j k_j(\mathbf{x}) = \sum_{j \in C_1} \lambda_j \sum_i z_{ij}x_i - \sum_{j \in C_2} \lambda_j \sum_i z_{ij}x_i - \sum_{j \in C_3} \lambda_j |\sum_i z_{ij}x_i| \quad (2)
\]

where \( z_{ij} = \mu_R(f_i, c_j) \) and the \( \lambda_j > 0 \) values specify the importances of the different characteristics. It is clear that by changing the \( \lambda_j \) values, the optimal solution of (2) is always a Pareto-optimal solution of the multiobjective problem (but it is not guaranteed that we can find every Pareto-optimal solution with this method).

4.1 Goal programming formulation

Problem (2) can be transformed into a linear optimization problem using a goal programming approach [23]. In this approach, we introduce new non-negative variables to measure the deviation of the objective for a given characteristic from the target value. According to the 3 classes of characteristics, we can specify the new variables as follows:

- if \( c_j \in C_1 \), then one new variable, \( d^+_j \geq 0 \) is introduced, and we want to maximize the positive deviation from the value 0 with the new constraint
  \[ \sum_{i=1}^{n} \mu_R(f_i, c_j)x_i - d^+_j = 0 \]

- if \( c_j \in C_2 \), then one new variable, \( d^-_j \geq 0 \) is introduced, and we want to maximize the negative deviation from the value 0 with the new constraint
  \[ \sum_{i=1}^{n} \mu_R(f_i, c_j)x_i + d^-_j = 0 \]

- if \( c_j \in C_3 \), then two new variables, \( d^+_j, d^-_j \geq 0 \) are introduced, and we want to minimize the deviation from the value 0 with the new constraint
  \[ \sum_{i=1}^{n} \mu_R(f_i, c_j)x_i - d^+_j + d^-_j = 0 \]

The obtained linear optimization problem can be formulated using these new constraints and variables as

\[
\max_{\mathbf{x} \in \{-1, 0, 1\}^n} \sum_{j \in C_1} \lambda_j d^+_j + \sum_{j \in C_2} \lambda_j d^-_j - \sum_{j \in C_3} \lambda_j (d^+_j + d^-_j)
\]

subject to

\[
\sum_{i=1}^{n} \mu_R(f_i, c_j)x_i - d^+_j = 0, \quad \forall c_j \in C_1
\]

\[
\sum_{i=1}^{n} \mu_R(f_i, c_j)x_i + d^-_j = 0, \quad \forall c_j \in C_2
\]

\[
\sum_{i=1}^{n} \mu_R(f_i, c_j)x_i - d^+_j + d^-_j = 0, \quad \forall c_j \in C_3
\]

\[
d^+_j \geq 0, \quad \forall j \in C_1 \cup C_3
\]

\[
d^-_j \geq 0, \quad \forall j \in C_2 \cup C_3
\]

(3)

5 Possible improvements of the model

5.1 More fragmented partitions

One possible extension of the model is the refinement of the partition of factors: instead of the feasible set \( \{-1, 0, 1\}^n \), we can use \( I^n_l \), where \( I_l \) is the set of integers with absolute value not greater than \( l \). For example, if \( l = 3 \), we can obtain a partition of factors into 7 subsets; these subsets can be associated with the terms:

- \( F_{-3} \) large decrease
- \( F_{-2} \) medium decrease
• $F_{-1}$ small decrease
• $F_0$ leave unaltered
• $F_1$ small increase
• $F_2$ medium increase
• $F_3$ large increase

Model 2 and 3 can be used for this purpose by changing only the feasible region of the problem.

5.2 Constraint on the number of factors to be changed

In practice, to increase/decrease the value of a factor can be associated with different cost values such as money and time. Since resources of a company are limited, the decision maker can specify an upper bound for the number of factors to be changed or specify the time and money available for the modifications. In the simplest case, in problem 2, the number of factors can be limited by using the constraint

$$\sum_{i=1}^{n} |x_i| \leq c, \quad (4)$$

where $c$ is the value given by the company.

Since Equation (4) is not linear in the $x_i$ variables, we need to reformulate it to include it in the model in Equation 3 and keep the linearity of this goal programming model. In order to do this, a set of binary variables has to be introduced: $x_i^+, x_i^- \in \{0, 1\}, i = 1, \ldots, n$. And the additional constraints to be included in the model are the following:

$$x_i - x_i^+ + x_i^- = 0, \forall i = 1, \ldots, n$$
$$x_i^+ + x_i^- \leq 1, \forall i = 1, \ldots, n$$
$$\sum_{i=1}^{n} (x_i^+ + x_i^-) \leq c \quad (5)$$

If there is information available about the specific resources required to change the values of factors (in terms of money or time), $c_i^+$ is the cost of increasing the value and $c_i^-$ is the cost of decreasing the value of factor $i$, and $C$ is the total budget, then we simply modify the last constraint of (5) as

$$\sum_{i=1}^{n} (c_i^+ x_i^+ + c_i^- x_i^-) \leq C \quad (6)$$

5.3 Model with uncertain parameters

As said before, set $L$ (representation of the relation) can be a set of fuzzy numbers, for example the set of all the triangular fuzzy numbers defined in the interval $[-1, 1]$. In this case the problem can be formulated in a similar way with the only difference that the parameters are fuzzy numbers instead of real numbers [3]. A very neat example of how to solve a linear programming problem with fuzzy parameters was offered in [8], where both the coefficients of the objective function and the parameters of the constraints can be expressed under the form of fuzzy numbers.

The formalization of the optimization problem is quite straightforward and thus we prefer to omit it. We only mention that there are mainly two approaches, one based on defuzzification and the other one based on the so-called aspiration levels. However, in general, a common feature and a crucial point in modeling optimization problems with fuzzy number is the choice of the ranking method [21].

6 Implementation in Excel

We implemented the goal programming formulation (Model 3) in Excel (see Fig. 5), with the additional option to specify an upper bound for the number of factors to be changed (the model with the more refined partition can be implemented in a similar way).

There are 17 different characteristics of the process/product associated with the machine under consideration: (runability, energy, md tensile, permeability, scott bond, formation, bulk, smoothness, brightness, opacity, density, gloss, absorption, two sided, dim. stab, friction, linting) and 24 factors divided into groups based on the processes (see Table 3).

To use the optimization model, the user has to specify first the partition of characteristics ($C_1$: to be increased; $C_2$: to be decreased; $C_3$: to leave unaltered), after that the importance of different characteristics and finally the maximal number of factors to be changed.

Example 2. As an example, we can specify the following partition of characteristics:

• $C_1$: runability, smoothness, density
• $C_2$: energy, absorption, permeability
• $C_3$: scott bond, formation, bulk, opacity, gloss

The importance of the characteristics in $C_1$, $C_2$ and $C_3$ are specified as 0.25, 0.25 and 1, respectively (which indicates that it is more important to leave the values in $C_3$ unaltered than to change the values in $C_1$ and $C_2$). The
Figure 5: Example of Excel spreadsheet. The main part represents the relation $R$, in the yellow part the user can model the query by partitioning set $C$. The last column on the right hand side yield the suggested partition of $F$. Green indicates that the factor should be increases, orange that it should be decreased and light blue that it should not be altered.

Table 3: Processes and factors connected to the paper machine

<table>
<thead>
<tr>
<th>Process</th>
<th>Factor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Raw material</td>
<td>DIP, (B)TMP%</td>
</tr>
<tr>
<td>Offset, roto</td>
<td>Filter, kaolin %</td>
</tr>
<tr>
<td>Offset</td>
<td>Filter, CaO%</td>
</tr>
<tr>
<td>Roto</td>
<td>Filter, talc</td>
</tr>
<tr>
<td>Stock prep.</td>
<td>DIP ret</td>
</tr>
<tr>
<td>HBX and forming</td>
<td>Flow rate</td>
</tr>
<tr>
<td>Pressing</td>
<td>Retention</td>
</tr>
<tr>
<td>Drying</td>
<td>Initial dryness, Initial drying rate</td>
</tr>
<tr>
<td>Calendering</td>
<td>Stack nip loadings, Steam amount</td>
</tr>
<tr>
<td>Reeling</td>
<td>Load</td>
</tr>
<tr>
<td>Winding</td>
<td>Winding</td>
</tr>
</tbody>
</table>

The maximum number of factors to be changed is 10. After running Excel Solver, the factors to be changed are the following:

- $F_1$ (increase): DIP, (B)TMP, Refining, Water in blade section, Retention, Steam amount (Pressing), Initial drying rate, Load (reeling)
- $F_2$ (decrease): GW, PGW; Filter, talc

If the factors are modified based on the optimal solution, then all the elements in $C_1$ will be increased, all the elements in $C_2$ will be decreased, and the characteristics in $C_3$ will be unaltered except for formation which will be increased slightly. This indicates that there exists no combination of factors which can result in exactly the required modifications but we can find one which satisfies the requirements for most of the characteristics.

7 Conclusions

In this study, we investigated how to construct an efficient decision support system in the context of paper industry. A (fuzzy) goal programming method is suggested to determine the optimal combination of factors (which describe the processes of a paper machine) in order to achieve the required goals (which are formulated using different characteristics of a paper machine). We transformed the original nonlinear multiobjective optimization problem into a single objective linear optimization formulation using goal programming approach. The model can be used to address many of the problems and issues asso-
associated with the management of the paper industry (and more specifically the effectiveness of paper machines) such as the need to increase the quality of the papers produced. The model was implemented in Excel and the use of the model was demonstrated through a numerical example. The results obtained indicate that the model is an efficient tool and can be used to help in making appropriate decisions regarding the complex system of paper machine processes. The approach discussed in the paper is reusable for other application domains described in terms of inputs and outputs.

**Acknowledgments**

This research has been funded through the TEKES strategic research project 40039/11.

**References**


