CHAPTER V

PRODUCTION PLANNING IN MANUFACTURING

INDUSTRY BY USING META–GOAL

PROGRAMMING

5.1 INTRODUCTION

Goal programming is a branch of multiobjective optimization, which in turn is a branch of multi-criteria decision analysis (MCDA), also known as multiple-criteria decision making (MCDM). This is an optimization programme. It can be thought of as an extension or generalisation of linear programming to handle multiple, normally conflicting objective measures. Each of these measures is given a goal or target value to be achieved. Unwanted deviations from this set of target values are then minimised in an achievement function. This can be a vector or a weighted sum dependent on the goal programming variant used.

Goal programming is used to perform three types of analysis:

- Determine the required resources to achieve a desired set of objectives.
- Determine the degree of attainment of the goals with the available resources.
- Providing the best satisfying solution under a varying amount of resources and priorities of the goals.

In short, GP underlies a realistic satisficing philosophy. Coherently with the satisficing philosophy in the GP models the deviations between the achievement of
the goals and their targets are minimized, that is, a certain function of the unwanted deviation variables is minimized (Ignizio [1983]; Romero [1991]).

Tamiz et. al. [1995] show how around 64% of GP applications reported in the literature minimize the unwanted deviation variables according to a lexicographic model, 21% according to a weighted model and the rest according to a minimax model (or a close relative like fuzzy GP).

It is clear that real world practitioners of Goal Programming never accept the first solution of a model as the definitive one. In this sense, some kind of sensitivity analysis is always carried out, taking into account some feedback from the decision-maker. Nevertheless, in many cases this sensitivity analysis is carried out using the same GP variant, while other parameters are changed. The purpose of this paper is to provide the users with different possible satisficing solutions using several variants at the same time using Meta goal programming model.

It will be shown how the approach can be more flexible than the usual GP models allowing to the decision-makers to establish target values not only for the goals but also for another criterion functions. In this sense, three types of meta goals are used:

**Type 1:** A meta-goal relating to percentage sum of unwanted deviations,

**Type 2:** A meta-goal relating to the maximum percentage deviation, and

**Type 3:** A meta-goal relating to the percentage of unachieved goals

Meta-GP approach can be very helpful to let the decision-maker clarify his/her knowledge of the problem situation and of his/her own preferences.
5.2 MATHEMATICAL MODEL

Let us consider the following general setting:

\[ f_i(x) + n_i - p_i = t_i, \quad i = 1, 2, \ldots, s \]

\[ g_j(x) \leq h_j, \quad j = 1, 2, \ldots, m \]

\[ x \in \mathbb{R}^n. \]  

(1)

We have a scenario with s goals plus m optional constraints, where functions \( f_i(x) \) are concave and functions \( g_i(x) \) are convex. Without loss of generality it is assumed that all the goals derive from attributes “more is better” what implies that the unwanted deviation variables are the negative ones. These unwanted deviation variables can be minimized following a lexicographic, weighted or minmax option. If the solutions obtained are not considered acceptable by the decision-maker then a mix of variants can be assayed. Alternatively, another possibility consists in the incorporation to the GP model of new aspiration levels. Let us clarify this idea. When using weighted option, the achievement function of the Goal Programming model can take the following form:

\[ h(n) = \sum_{i=1}^{s} w_i \frac{n_i}{t_i} \]  

(2)

where \( w_i \) represents a preferential weight, and the deviation variables have been normalized, by dividing them among their corresponding target values. If the target value is zero or negative, then the unwanted deviation variable should be normalized by resorting to any other system like the Euclidean normalization (see Tamiz et. al. [1998], pp. 572-573). In this case, its optimal value can be interpreted as the minimum percentage sum of weighted unwanted deviation variables that can
be achieved. On the other hand, if the minmax variant is chosen, the optimal value of the achievement function

\[
\max_{i=1,2,...,s} \left\{ \frac{w_i n_i}{t_i} \right\}
\]

(3)

can be interpreted as the minimum maximum percentage weighted deviation from the target values which can be achieved.

Given these facts, it will now be assumed that the decision-maker may want to give aspiration levels for the final values of these achievement functions. This originates new sets of goals that are, in some sense, goals of the original goals. That is why they are called meta-goals. In a first approach; the following types of meta-goals can be defined:

**Type 1:** The percentage sum of unwanted deviation variables cannot surpass a certain bound \(Q_1\). This desire of the decision-maker means to impose the following constraint:

\[
\sum_{i=1}^{s} w_i \frac{n_i}{t_i} \leq Q_1
\]

(4)

**Type 2:** The maximum percentage deviation variables cannot surpass a certain bound \(Q_2\). This desire of the decision-maker means to impose the following set of constraints:

\[
\max_{i=1,2,...,s} \left\{ \frac{w_i n_i}{t_i} \right\} \leq Q_2 \iff \left\{ \begin{align*}
\frac{w_i n_i}{t_i} &- D \leq 0, & i = 1, 2, ..., s, \\
D &\leq Q_2,
\end{align*} \right.
\]

(5)

where \(D\) represents the maximum percentage weighted deviation.
Apart from these two types of meta-goals, another case can be considered, which do not directly take into account achievement functions of Goal Programming models. Thus, the following type of goals seems relevant.

**Type 3:** The percentage of achieved goals cannot surpass a certain bound $Q_3$. This desire of the decision-maker can be modeled in the following way:

$$n_i - M_i y_i \leq 0, \quad i = 1, 2, \ldots, s$$

$$\sum_{i=1}^{s} y_i \leq Q_3$$

(6)

where $y_i$ are binary variables and $M_i$ represent arbitrarily large values that the corresponding attributes cannot achieve. Consequently the value of

$$\sum_{i=1}^{s} y_i$$

in the optimum solution measures the number of goals that have not been fully achieved.

In a more general formulation, it can be assumed that the decision-maker may wish to establish meta-goals of any of the previously defined types, but involving only certain subsets of the full set of goals. More precisely, let us suppose that a type 1 meta-goal is imposed on the set

$$S_k^{(1)} = \{1, 2, \ldots, s\}.$$ 

Then, the goal takes the form

$$\sum_{i \in S_k^{(1)}} w_i \frac{n_i}{t_i} \leq Q_k^{(1)}.$$ 

(7)

Meta-goals on achievement functions corresponding to certain priority levels of a lexicographic scheme can be incorporated into this scheme in a natural way.
Similarly, for a type 2 meta-goal on the set $S_r^{(2)}$ and type 3 meta goal on the set $S_r^{(3)}$, we have

$$w_i \frac{n_i}{t_i} - D_i \leq 0, \ i \in S_r^{(2)}, \ D_i \leq Q_i^{(2)} \tag{8}$$

and

$$n_i - M_i y_i, i \in S_r^{(3)}$$

$$\sum_{i \in S_r^{(3)}} \frac{y_i}{\text{card } S_r^{(3)}} \leq Q_i^{(3)}$$

$$y_i \in [0,1], i \in S_r^{(3)} \tag{9}$$

respectively.

In most of the situations the above constraints define an empty feasible set. For instance, the existence of a feasible solution for the last set of constraints will mean that there is a GP solution for which all the targets are fully achieved what it is extremely unlikely. Even, if a feasible solution exists it will likely to be non-efficient (see, for example Tamiz and Jones, 1996; Caballero et al., 1996). A possible way to redeem this situation obtaining a satisficing solution consists in formulation a GP model of the above GP setting. Namely, let us suppose that the decision-makers set $r_1$ type 1 meta-goals, $r_2$ type 2 meta-goals, $r_3$ type 3 meta-goals. In this way, the following Meta-GP or $[GP]^2$ model is proposed.

$$\text{Min } \beta^{(1)}_1, \ldots, \beta^{(1)}_{r_1}, \beta^{(2)}_1, \ldots, \beta^{(2)}_{r_2}, \beta^{(3)}_1, \ldots, \beta^{(3)}_{r_3}$$

such that

$$f_i \cdot x_i + n_i - p_i = t_i, \quad i = 1, 2, \ldots, s$$

$$g_j \cdot x_j \leq b_j, \quad j = 1, 2, \ldots, m$$
\[
\sum_{i \in S_l} w_i \frac{n_i}{l_i} + \alpha_k^{(1)} - \beta_k^{(1)} = q_k^{(1)}, \quad k = 1, 2, \ldots, r
\]

\[
w_i \frac{n_i}{l_i} - D_i \leq 0, \quad i \in S_l^{(2)}, \quad l = 1, 2, \ldots, r
\]

\[
D_l + \alpha_i^{(2)} - \beta_l^{(2)} = q_i^{(2)}, \quad i = 1, 2, \ldots, r
\]

\[
n_i - M_i y_i, \quad i \in S_i^{(3)}, \quad l = 1, 2, \ldots, r
\]

\[
\sum_{i \in S_i^{(3)}} \frac{y_i}{\text{card } S_i^{(3)}} + \alpha_i^{(3)} - \beta_i^{(3)} = q_i^{(3)}, \quad r = 1, 2, \ldots, r

y_i \in 0, 1, \quad i \in S_i^{(3)}, \quad r = 1, 2, \ldots, r
\]

\[
n_i, \quad p_i \geq 0, \quad i = 1, 2, \ldots, s
\]

It is obvious that the achievement function of this model can adopt any of the alternative formulations (i.e., weighted, lexicographic, minmax, etc).

### 5.3 AN ILLUSTRATIVE EXAMPLE OF MGP MODEL

A furniture manufacturer produces five kinds of products, chair, bench, table, sofa, and bed. The production of all products is done in two separate machine centres within the plant. Each chair requires 2 hrs in machine center 1 and 1 hr in machine center 2, each bench requires 1 hr in machine center 1 and 2 hr in machine center 2, each table requires 1 hr in machine center 1 and 3 hr in machine center 2, each sofa requires 4 hrs in machine center 1 and 1 hr in machine center 2. Each chair requires 3 hrs in machine center 1 and 1 hr in machine center 2. The gross margin
from the sale of a chair is Rs. 100, from bench Rs. 500, from table Rs. 50, from sofa Rs. 100 and from bed Rs. 150. In addition, each product requires some in-process inventory. The per-unit in-process inventory required is Rs. 50, 30, 30, 20, 20 respectively for all five products.

The firm has normal monthly operation hours of 360 for machine center 1 and 240 for machine center 2. According to the marketing department, the forecasted sales for all five products is 150, 100, 120, 60, 10 units respectively for the coming month.

The plant manager has established the following goals for production in the next month:

(1) Earn a gross profit of Rs. 50000 in the next week.

(2) Limit the amount tied up in in-process inventory for the month to Rs. 10,000.

(3) Achieve the sale goal of all five products.

(4) Avoid any underutilization of regular operation hours of both machine centers.

(5) Limit the overtime operation to 100 hrs for each machine center 1 and 2.

Assuming the three groups are of equal importance, the following weighted goal programme is formulated:

5.4 FORMULATION OF MODEL

Let

\[ X_i = \text{Number of units of product to be sold; } i = 1, 2, 3, 4, 5 \]
\( n_i = \text{Underachievement of sales goal of product } i \)

\( p_i = \text{Overachievement of sales goal of product } i \)

\( n_6 = \text{underutilization of regular operation hours of machine center 1} \)

\( p_6 = \text{Overachievement of regular operation hours of machine center 1} \)

\( n_7 = \text{underutilization of regular operation hours of machine center 2} \)

\( p_7 = \text{Overachievement of regular operation hours of machine center 2} \)

\( n_8 = \text{Underachievement of amount tied up in in-process inventory for the month} \)

\( p_9 = \text{Overachievement of amount tied up in in-process inventory for the month} \)

\( n_{10} = \text{underutilization of overtime operation hours of machine center 1} \)

\( p_{10} = \text{Overachievement of overtime operation hours of machine center 1} \)

\( n_{11} = \text{underutilization of overtime operation hours of machine center 2} \)

\( p_{11} = \text{Overachievement of overtime operation hours of machine center 2}. \)

Thus, Weighted Goal Programming Problem is defined as:

\[
\text{Minimize: } n_1 / 150 + n_2 / 100 + n_3 / 120 + n_4 / 60 + n_5 / 10 + n_6 / 360 + n_7 / 240 \\
+ p_8 / 10000 + n_9 / 50000 + p_{10} / 100 + p_{11} / 100
\]

\text{Subject to}

\[
x_1 + n_1 \quad p_1 = 150 \\
x_2 + n_2 - p_2 = 100 \\
x_3 + n_3 \quad p_3 = 120 \\
x_4 + n_4 - p_4 = 60 \\
x_5 + n_5 \quad p_5 = 10 \\
2 x_1 + x_2 + x_3 + 4 x_4 + 3 x_5 + n_6 - p_6 = 360
\]
Using the goal programming methodology, the solution is:

\[
x_1 = 40, \ x_2 = 100, \ x_3 = 10, \ x_4 = 60, \ x_5 = 10,
\]
\[
n_1 = 110, \ p_1 = 0, \ n_2 = 0, \ p_2 = 0, \ n_3 = 110, \ p_3 = 0, \ n_4 = 0, \ p_4 = 0,
\]
\[
n_5 = 0, \ p_5 = 0, \ n_6 = 0, \ p_6 = 100, \ n_7 = 0, \ p_7 = 100, \ n_8 = 3300, \ p_8 = 0,
\]
\[
n_9 = 0, \ p_9 = 12000, \ n_{10} = 0, \ p_{10} = 0, \ n_{11} = 0, \ p_{11} = 0.
\]

Objective function (Total deviation): 1.65

Let us assume that the decision-maker does not consider acceptable the above solution. Thus the above weighted goal programme is extended to a meta-goal programme with the following three meta-goals:

**MG1:** The percentage maximum deviation from all goals should be at most 60%

**MG2:** The maximum percentage deviation from any goal should be at most 40%

**MG3:** At most seven goals should be unsatisfied.

Their relative importance is given in the table below:

<table>
<thead>
<tr>
<th>Meta-goal</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>MG1</td>
<td>4.0</td>
</tr>
</tbody>
</table>
Thus, the Meta GP formulation is given as:

**Minimize:** $4.0 \left( b_1 / 0.6 \right) + 2.0 \left( b_2 / 0.4 \right) + (b_3 / 7)$

**Subject to**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>MG2</td>
<td>2.0</td>
</tr>
<tr>
<td>MG3</td>
<td>1.0</td>
</tr>
</tbody>
</table>

\[
x_1 + n_1 \quad p_1 = 150
\]
\[
x_2 + n_2 - p_2 = 100
\]
\[
x_3 + n_3 \quad p_3 = 120
\]
\[
x_4 + n_4 - p_4 = 60
\]
\[
x_5 + n_5 \quad p_5 = 10
\]
\[
2 x_1 + x_2 + x_3 + 4 x_4 + 3 x_5 + n_6 - p_6 = 360
\]
\[
x_1 + 2 x_2 + 3 x_3 + x_4 + x_5 + n_7 - p_7 = 240
\]
\[
50 x_1 + 30 x_2 + 30 x_3 + 20 x_4 + 20 x_5 + n_8 \quad p_8 = 10000
\]
\[
100 x_1 + 500 x_2 + 50 x_3 + 100 x_4 + 150 x_5 + n_9 - p_9 = 50000
\]
\[
p_6 + n_{10} \quad p_{10} = 100
\]
\[
p_7 + n_{11} - p_{11} = 100
\]
\[
z_1 - (n_1/150 + n_2/100 + n_3/120 + n_4/60 + n_5/10 + n_6/360 + n_7/240 +
p_8/10000 + n_9/50000 + p_{10}/100 + p_{11}/100) = 0
\]
\[
-z_2 + D = 0
\]
\[
n_1 - 150D \leq 0
\]
\[
n_2 - 100D \leq 0
\]
\[
n_3 - \ldots \leq 0
\]
\[ n_4 - 60D \leq 0 \]
\[ n_5 - 10D \leq 0; \]
\[ n_6 - 360D \leq 0 \]
\[ n_7 - 240D \leq 0 \]
\[ p_8 - 1000D \leq 0 \]
\[ n_9 - 50000D \leq 0 \]
\[ p_{10} - 100D \leq 0 \]
\[ p_{11} - 100D \leq 0 \]
\[ n_1 - 1500 y_1 \leq 0 \]
\[ n_2 - 1000 y_2 \leq 0 \]
\[ n_3 - 1200 y_3 \leq 0 \]
\[ n_4 - 600 y_4 \leq 0 \]
\[ n_5 - 100 y_5 \leq 0 \]
\[ n_6 - 3600 y_6 \leq 0 \]
\[ n_7 - 2400 y_7 \leq 0 \]
\[ p_8 - 100000 y_8 \leq 0 \]
\[ n_9 - 500000 y_9 \leq 0 \]
\[ p_{10} - 1000 y_{10} \leq 0 \]
\[ p_{11} - 1000 y_{11} \leq 0 \]
\[ z_3 + (y_1 + y_2 + y_3 + y_4 + y_5 + y_6 + y_7 + y_8 + y_9 + y_{10} + y_{11}) = 0 \]
\[ z_1 + a_1 - b_1 = 0.60 \]
\[ z_2 + a_2 - b_2 = 0.40 \]
\[ z_3/11 + a_3 - b_3 = 7/11 \]
where \( y_i = \begin{cases} 
1, & \text{if goal } i \text{ is not satisfied} \\
0, & \text{otherwise} 
\end{cases} \), \( i = 1, 2, ..., 11 \).
programming, that of combination of and trade-off between underlying distance metrics and hence philosophies. Thus, this approach could be used as a second stage after a traditional GP problem has been solved.

This approach can be considered like a “sensitivity analysis” of the solutions obtained as well as of the own model structure. This scheme requires a certain interaction with the decision-maker in order to adjust the target values of the meta-goals, but it is much more flexible than usual GP formulations and besides, it lets the decision-maker to clarify his/her knowledge about his/her actual structure of preferences.

The proposed approach can be applied to more realistic problems with non-linear goals and/or non-continuous variables.