ANNALS of the ORADEA UNIVERSITY.

Fascicle of Management and Technological Engineering, Volume VI (XVI), 2007

THE OPTIMIZING METHODS FOR FLEXIBLE STRUCTURES

FOTA ADRIANA
Transilvania University of Brasov

Key words: flexible manufacturing system, optimizing, modeling

Abstract: Most of the complex problems, which appear in the manufacturing domain, and the management of manufacturing systems (flexible manufacturing systems) are optimizing problems. Mathematical modeling of optimizing problems offers the opportunity to find the optimal solution (solutions), with immediate consequences for the system's economical efficiency increasing. To optimize is the action of obtaining best results for some given circumstances. In this paper is presented the principal methods for optimizing flexible manufacturing systems and a fuzzy model which shall contain a linguistic equation group (group of rules), model that is used in achieving the functioning algorithm.

1. INTRODUCTION

A flexible manufacturing system is a manufacturing system, which there is some amount of flexibility, which allows the system to react in the case of changes, whether predicted or unpredicted. This flexibility is generally considered to fall into two categories, within which are numerous other subcategories. The first category, machine flexibility, and covers the system's ability to be changed to produce new product types, and ability to change the order of operations executed on a part. The second category of flexibility within an FMS is called routing flexibility, which consists of the ability to use multiple machines to perform the same operation on a part, as well as the system's ability to absorb large-scale changes, such as in volume, capacity, or capability.

A central computer commonly controls the whole FMS.

The main advantages of a FMS are its high flexibility in managing manufacturing resources like time and effort in order to manufacture a new product. The best application of a FMS is found in production of small sets of products that are likely but not equal that those from a mass production, otherwise production cost of small sets of products will cost a lot in relation with mass production cost.

A flexible manufacturing system (FMS) is a production system consisting of a set of identical and/or complementary numerically controlled machines, which are connected through an automated guided vehicle system. Since FMS is capable of producing a variety of part types and handling flexible routing of parts instead of running parts in a straight line through machines, FMS gives great advantages through the flexibility, such as dealing with machine and tool breakdowns, changes in schedule, product mix, and alternative routes. Flexible manufacturing is of increasing importance in advancing factory automation that keeps a manufacturer in a competitive edge.

While FMS offers many strategic operational benefits over conventional manufacturing systems, its efficient management requires solution to complex process planning problems with multiple objectives and constraints. The aim of process planning is to develop a cost effective and operative process plan over the planning phases. Decisions regarding the process-planning problem have to be made before the start of actual production, and consist of organizing the limited production resource constraints efficiently. Generally, the process planning includes routing optimization, equipment optimization and machine optimization.

Because in any given real situation, the necessary effort and the wanted benefit can be expressed in terms of a function with well defined decision variables, optimizing can be defined as finding the conditions which give the function has minimum or maximum values. Optimal searching methods are known as mathematical programming functions and are studied as parts of operational research.

2. OPTIMIZING METHODS FOR FLEXIBLE STRUCTURES

2.1 Multi-objective optimizing

The optimizing process uses a multitude of initial values and corresponding solutions, which are user defined or random generated by the algorithm. Optimizing algorithm generates then successive these kind of values, leading to an optimal one. When more than one variant are possible, the best possible solution is to be chosen.

Mathematical modeling of optimizing problems offers the opportunity to find the optimal solution (solutions), with immediate consequences for the system's economical efficiency increasing. These problems are resolved using mathematical programming. Multi-objective optimizing is a matter of study for mathematical programming. This type of optimizing starts with the premise that the number of potential solutions is infinite, by solving it, the optimal solution (solutions) is determined.

For the general formulation of the problem, it is considered that a n-dimensional vector x_i is applied to the input of the system, a m-dimensional x_e vector is obtained from the output and the parameters of the systems are described by p-dimensional vector q.

Mathematical programming is a maximizing or minimizing problem for one or more functions defined by variables, called objective-function (functions), purpose-function (functions) or efficiency-function (functions), whose variables satisfy a system of restrictions expressed by equalities or inequalities.

Objective-function is also a r-dimensional vector of r functions. Having the linking relations between input, output and parameters and also the limiting conditions imposed, the overall shape of *the multi-objective optimizing* is the following:

$$R_k(x_i\ , x_e\ , q, t) \le 0,\ \ k=1,\dots,k \qquad (1.1)$$
$$\text{optimum}\ [\ f_1(x_i\ , x_e\ , q\ , t),\dots,\ f_r(x_i\ , e_e\ , q\ , t)]$$

In the above written relations *t* means the time, and *k* is the total number of linking and conditioning relations between mentioned values. These relations are called *restrictions*. Requested optimizing for the second expression is usually obtained by minimizing some of those *r* components and maximizing other.

If both restrictions and objective-functions are linear functions in relation with variables and also are time independent, then the problem becomes a multi-objective linear one.

When the objective function vector has a single component it leads to a linear programming mono-objective problem.

The model for mathematical programming problem [1, 2] is the following:

$$\max\ (\min)(z_h=f_h(x_1,x_2,\dots,x_n)),\ h=1,\dots,\ r$$
$$g_i(x_1,x_2,\dots,x_n)\rho_i,\ \ i=1,\dots,\ m$$
$$\rho_i\in\{\le,\ =,\ \ge\ i=1,\dots,\ m \qquad (1.2)$$

If *r>1*, the relation above is multi-dimensional; *r=1* the objective function has a single component => **linear mono-objective programming**. When the objective functions f_h (h=1,…,r) and g_i (i=1,…,m) are linear, the (1.2) model is named ***linear***.

The mathematical model is made from three parts:

- Input parameters, which values are known;
- Output parameters, given by processing;
- Functional relations between input and output values, algebraic equations, deferential equations, integral equations, etc.

For resolving linear mono-objective programming problems we can use solving algorithms, the most known is SIMPLEX. The SIMPLEX algorithm is a systemic and economic method for basic programming exploring (precisely the transit between basic programs to the advanced ones), that the objective function values always improve, until it reaches the optimal. The algorithm gives criteria for the case in which the linear programming problem has no program or has infinite optimal ones.

The optimizing models with representative restrictions (mathematical programming), with multiples applications possibilities into techniques, economics, socials systems and processes optimizing, there are: linear models, discreet models, bivalent models, convex models, quadratic models, transport models, special models for non-linear programming. By using linear programming, can be solved manufacturing domain problems like: optimal organization of a production sector, plant layout problems, choosing the optimal technologies problems, manufacturing and supply problems, investments problems, the optimal capability of production equipments problems, etc.

If the objective function given in the (1.2) model are non-linear with respect to the unknown ones and the objective function vector has a single component, then the problem become a *non-linear programming*. A lot of problems, even the ones mentioned above as accessible by linear programming, they all have a linear attribute. This category includes also the flexible manufacturing systems problem.

For describing technical, economical, etc. processes which develop in time is used *dynamic programming model,* [3], which is based on Bellman's optimal principle:

- Assuming that the time element $[t_0, t_f]$ divides in steps, so the system state changes by passing over one stage to another, depending on external decisions of the system. Inside each step it accepts only a decision;
- the system status is analyzed and described by a status vector $x=(x^1,x^2,\ldots,x^m)\in R^m$, the decision by a decision vector $u=(u^1,\ldots,u^r)\in R^r$. The components of these vectors are named status values (x^i), decision values (u^i).

The dynamic programming determines a sequence of decisions $\left(u_1^*,u_2^*,\ldots,u_n^*\right)$ and a sequence of corresponding states $\left(x_0^*,x_1^*,\ldots,x_n^*\right)$ which gives the *optimal value* for total evaluation (of the objective-function).

The dynamic programming determines a sequence of decisions $\left(u_1^*,u_2^*,\ldots,u_n^*\right)$ and a sequence of corresponding states $\left(x_0^*,x_1^*,\ldots,x_n^*\right)$ which gives the ***optimal value*** for total evaluation (of the objective-function).

The mathematical model is the following:

$$\text{opt}\sum_{j=1}^{n} g_j\left(x_{j-1},u_j\right);x_0\text{-dat}\; x_j = f_j\left(x_{j-1},u_j\right);\ j=1,n;\ x_j \in \sum\nolimits_j\ ,j=1,\ldots,n;\ u_j \in \Omega_j\left(x_{j-1}\right),\ j=1,n \quad (1.3)$$

Where fj, g_j - objective functions; x_j - possible states from the states domain $\sum_j \subseteq R^m$; u_j - possible decisions from decisions domain $\Omega_j\left(x_{j-1}\right)\subseteq R^r$.

The evolution of the system (process) is shown bellow (Figure 1.1):

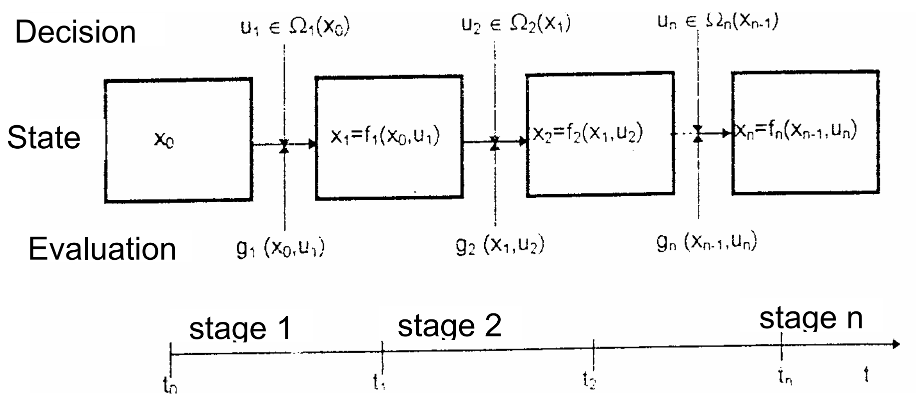

Fig. 1.1. The evaluation of the process

2.2 Multi-attribute optimizing. The fuzzy model

Multi-dimensional decisions have many criteria and it is based on mathematical programming with many objective functions.

There are different ways to approach multi-attribute problems [4]: utilities theory (analyses the way in which activity is conditioned by the independence property of criteria); methods for the optimal alternative choice, which takes into account both the favorable consequences arguments and the unfavorable ones; mathematical programming methods having multiple optimal criteria.

Fuzzy measures of the (degree of) satisfaction of each one of the three objectives are taken for each feasible alternative. These measures are weighted according to the importance of each criterion, obtaining a “score” for the given alternative. This “score” represents the degree of satisfaction of the overall objective by a certain alternative, and is given by:

$$\mu_0(x_k) = [\mu_{c_1}(x_{k_1})]^{\alpha_1} \cdot [\mu_{c_2}(x_{k_2})]^{\alpha_2} \cdot [\mu_{c_3}(x_{k_3})]^{\alpha_3} \quad (1.4)$$

In (1.4) $\mu_0(x_k)$ is the overall objective degree of satisfaction corresponding to the k-th alternative, $\mu_{c_i}(x_{k_i})$ is the degree of satisfaction of the i-th objective (relatively to the k-th alternative) and α_I its weight. The alternative corresponding to the highest overall objective degree of satisfaction is chosen. The importance of each criterion is given by the weights obtained from a pair wise comparison matrix through the λ_{max} technique. The pair wise comparison matrix (in this case 3 x 3) usually contains human expert linguistic estimates of pair wise comparisons between the objectives importance. This decision structure is completely defined once the pair wise comparison matrix and the membership functions for each objective are given. The membership function for low workload and low processing time, is piecewise linear at first and then exponential, and is completely defined (for any objective) by three parameters. The membership function for low distance is a discrete one and it is arbitrarily assigned. It will be assumed that experts already specify the pair wise comparison matrix, e.g..

A specific algorithm of treating the information features the fuzzy modeling.

Fuzzy systems are processing information according to an own philosophy, carrying out of principle on grounds of the following flow:

{input variables} ⇒ (fuzification) ⇒ (interference) ⇒ (composition) ⇒ (defuzification) ⇒{output variables}.

In view of structural analysis of a FMS for round shafts processing, former [2, 4] the decomposition of the system into component sub-systems was carried out, connections among these and the transfer function were established. On grounds of the structural decomposition draft the fuzzy model may be elaborated [2], which shall contain a linguistic equation group (group of rules), model that is used in achieving the functioning algorithm. Finally for the connected sub-systems within the FMS for round shafts processing, the final group of rules shall be generated, out of which the system outputs may be extracted. The program is written as a set of the rules due to each sub-system (work stations, robots, conveyors, stocks).

The fuzzy model is elaborated on macro-level, for connections between adjacent sub-systems. Fuzzy logic is a general conclusion of the classic, bivalent logic, replacing its discrete character in (0, 1) with one of continuous nature. The fundamental fuzzy logic is made of the multivalent logic. So as for the deterministic bivalent logic "1" is associated to TRUE and "0" is labeled FALS, in the fuzzy logic for a deterministic positive real number variable, the associated linguistic variable may have linguistic degrees: BIG, AVERAGE, SMALL. Because of the expression by linguistic variables, mathematical modeling by fuzzy logic may be easily approached within the complex structure study, such as FMS for round shafts processing.

A fuzzy rule appears when it exist a premise concerning the event, which implies a certain logic consequence (conclusion): IF (conditions, restrictions) THEN (effect / consequences)............... ELSE (consequences / risks).

The fuzzy rule base is built up by putting fuzzy multitudes associates to output variables, in logic contact with fuzzy multitudes of the input variables [2]. The fuzzy rule group modeling as linguistic equations the FMS for round shaft processing function is presented further on.

1. *Fuzzy rules for state estimation.* In the case of MISO (Multi Input Single Output) – type systems, the group of rules shows like:

R1: IF X IS A1…& Y IS B1 THEN Z IS C1

R2: IF X IS A2…& Y IS B2 THEN Z IS C2

⋮

Rn: IF X IS An…& Y IS Bn THEN Z IS C1 (1.5)

where: X, Y, Z are linguistic variables, representing the system state. Ai, Bi, Ci are linguistic values of the linguistic variables x, y and z. A more general shape is that, where consequences are represented by a function having as variables the process states:

Rt: IF X IS At,…,& Y IS Bt THEN $Z = f_t(x,..,y)$ (1.6)

and t represents the evolution of the system at different moments.

2. *Fuzzy rules for object estimation.* These rules derive from the experience of the human operator and refer to the fuzzy control rules of the object:

Rt: IF X IS At,…,& Y IS Bt THEN $Z = f_t(x,..,y)$ (1.7)

This set of rules was written on grounds of the functional connections between sub-systems (inputs – x_{pi} and outputs – y_{qi}) established within the decomposition draft and coupling matrixes between sub-systems, former elaborated [3].

The set of rules was drawn up for processing three round shaft families: FR_1 (compact round shaft family) – of high complexity; FR_5 (threaded shaft family) – average complexity and FR_4 (family of axles and spindles) – law complexity. The set of rules for system stockings also was elaborated [3].

Further on selectively, sequences of the functioning algorithm of the analyzed system for processing a more complex item family – FR_1, are presented, similarly being elaborated also the algorithm for the other five item families FR_2 ... FR_6 processed within the flexible manufacturing system of round shafts processing.

For example, the rule group for processing the item family FR_1 – compact round shafts is as follows, [3]:

T001 IF $\exists R_1$ AND$x_{33}=y_{21}\&K_{23}=1$AND $x_{31}=y_1\&K_{18}=1$ THEN $y_{33}=1R_1$ ELSE $y_{33}=1\ R_2$ OR $y_{33}=1R_3$

T002 IF $x_{45}=y_{33}\ \&\ K_{34}=1$ AND $x_{46}=y_{32}\ \&\ K_{34}=1$ AND $x_{41}=y_{12}\ \&\ K_{14}=1$ AND $x_{42}=y_{53}\ \&\ K_{54}=0$ A ND $x_{43}=y_{52}\ \&\ K_{54}=0$ AND $x_{44}=y_{13,8}\ \&\ K_{13,4}=0$ THEN $y_{43}=1$

T003 IF $x_{53}=y_{43}\ \&\ K_{45}=1$ AND $x_{52}=y_{42}\ \&\ K_{45}=1$ AND $x_{51}=y_{13}\ \&\ K_{15}=1$ THEN m_{FC1} OR m_{FC2} OR … OR $m_{FCp}=1$

T004 IF $T_o=\Delta t=1s$ AND $x_{51}=y_{13}\ \&\ K_{15}=1$ AND m_{FC1} OR m_{FC2} OR … OR $m_{FCp}=0$ THEN $y_{52}=1$

T005 IF $x_{42}=y_{53}\ \&\ K_{54}=1$ AND $x_{43}=y_{52}\ \&\ K_{54}=1$ AND $x_{41}=y_{12}\ \&\ K_{14}=1$ AND $x_{46}=y_{32}\&K_{34}=1$ AND $x_{45}=y_{33}\ \&\ K_{34}=0$ AND $x_{44}=y_{13,8}\ \&\ K_{13,4}=0$ THEN $y_{44}=1$

⋮

T051 IF $x_{52,1}=y_{1,41}\ \&\ K_{1,52}=1$ AND $x_{52,2}=y_{50,4}\ \&K_{50,52}=1$AND $x_{52,3}=y_{50,3}\ \&\ K_{50,52}=1$THEN $y_{52,3}=1$

3. CONCLUSION

Optimizing is build from algorithms in which properties can be quantified in the terms of functions. The optimizing process uses a multitude of initial values and corresponding solutions, which are user defined or random generated by the algorithm. Optimizing algorithm generates then successive these kind of values, leading to an optimal one. When more than one variant are possible, the best possible solution is to be chosen.

REFERENCES

[1] Abrudan, I., Sisteme flexibile de fabricație. Concepte de proiectare şi management. Editura Dacia, Cluj – Napoca, 1996.

[2] Boncoi Gh., Calefariu G., Fota A., Sisteme de producție, vol. I, II Editura Universității Transilvania din Brasov, Romania, 2000.

[3] Fota A., Proiectarea sistemelor de maşini. Modelare şi simulare, Editura Universității Transilvania din Brasov, Romania, 2004.

[4] Fota A., Cercetări privind dimensionarea şi configurarea sistemelor flexibile de fabricație pentru prelucrarea arborilor circulari, Teza de doctorat, Transilvania University of Braşov, Romania, 2003.

[5] Mohora, C., Simularea sistemelor de producție, Editura Academiei Române, Bucureşti, 2001.