# Design and Implementation of a Genetic-Fuzzy-Evolutionary System for Manufacturing Planning and Scheduling

Rodolfo E. Pérez-Loaiza [1] and Carlos Alberto Reyes-García [2]

[1] Instituto Tecnológico de Apizaco-COSNET, Universidad Metropolitana de Puebla
Av. Instituto Tecnológico s/n, Apizaco, Tlax., 90300. México

[2] Instituto Nacional de Astrofísica Óptica y Electrónica, INAOE
Luis E. Erro # 1, 72840 Tonantzintla, Pue., México

*Abstract:* This paper presents the process of design and implementation of a hybrid system which is able to perform the required operations to dynamically determine the maximum production capacity, inside a factory plant, according to the production or sales plans, at any given period. This task is done along with optimizing the assignment of the different products, to the production lines. The complete system consists on the implementation of a genetic algorithm which uses a particularly designed fuzzy fitness function, and an evolutionary programming algorithm. The designed architecture and the implementation process are presented, as well as several results from tests applied to some real productive processes.

*Keywords:* Hybrid Systems, Planning and Scheduling, Fuzzy-Genetic Algorithms, Evolutionary Algorithms, Production Optimization.

## 1 Introduction

In the manufacturing industries there are planning trends looking to reach the greatest possible integration of all the different variables acting inside them. Some of these trends are represented by systems like; Just In Time (JIT), Computer Integrated Manufacturing (CIM), and the MRP II system, among others[4]. Nowadays, the sequential manufacturing planning, the traditional scheduling programs, as well as the control strategies are being insufficiently flexible to respond to the continuous production style changes, and to the highly dynamic variations in the products requirements [4]. What now is needed is the development of systems not only highly productive but rather, also, very flexible. It is important to mention that there are many engineering models with the ability to approach these planning problems, for example: mathematical programming, linear programming or the Simplex approach, and dynamic programming, to mention some. In most of these methods a set of equations should be solved in order to find a feasible basic succession of solutions, each one better than the previous one, until a good solution is found. These methods are adequate when the function to optimize is linear, and the variables that control the problem are few. However, if the problem is not linear, and/or the number of intervening variables is large, the problem gets highly complex. Mainly in those problems that cannot even be expressed by a mathematical function. For this reason, it is necessary to appeal to computer tools to carry out a large amount of repeated calculations. And even so, the solutions are very distant of being the optimums [3]. In this way, there is a need for new tools to allow the dynamic and flexible adaptation of production plans, required to survive in the modern aggressive competitive market. With this idea in mind, we came up with the design of the following hybrid system.

## 2 Hybrid System Architecture

The system is conformed mainly by 5 modules, which are related in an orderly sequence to determine its correct operation and the easy interaction with the user, in such a way that the wanted results might be obtained. Figure 2.1 shows the structure representing the schematic operation of the proposed intelligent system.

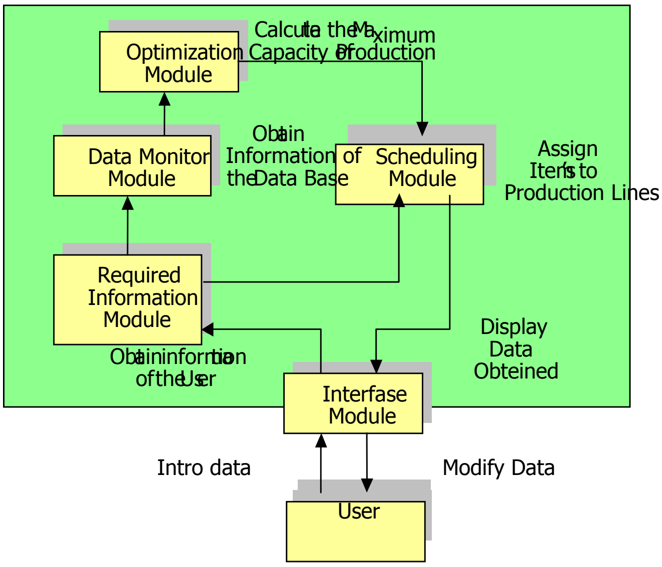



Fig. 2.1 General architecture of the Hybrid System

The system starts with the market demand or sales orders data, which will form the Production or Sales Plan for a given period. As well as with the information corresponding to the production lines planned to use. At the same time, the system obtains the necessary information from the different databases, to carry out the needed operations. One of them is the determination of the present maximum production capacity, with the objective of finding out whether the production or sales plan can be completed in the predicted time. If the answer is positive, the possible maximum amount of each product to be produced is calculated [1]. To perform this task we designed a genetic algorithm [5] whose fitness function is fuzzy.

The last stage, which consists mainly on the optimum assignment of the products to manufacture at each production line, is carried out by means of a designed evolutionary algorithm [8]. With the use of this algorithm, the minimum dead time and the minimum production required time at each production line are obtained.

# 3 System Modules

## 3.1 Interface Module

This module allows the interaction of the user with all the other modules in the system, in a friendly manner.

## 3.2 Required Information Module (System Inputs)

This module handles information mainly coming from three sources:

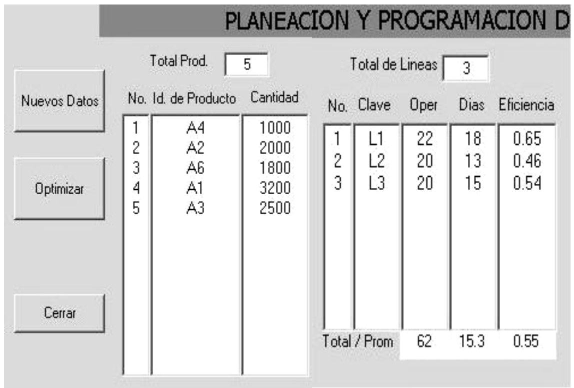



Fig. 3.1 Capturing the Production or Sales Plan

**Production or Sales Plan:** Basically, this plan consists on establishing the demand figures that the company should reach to complete the goals of the company, according to an established time [2]. It also indicates the quantities to be manufactured of each one of the products requested by the clients, as well as the information on the production lines where they will be produced, (see Figure 4.1).

**Databases:** Contain the company's information required by the system for all the production operations.

**User Modified Data:** It consists on the changes on the data introduced by the user. The modification may be due to sudden changes in the Production or Sales Plan, caused by new orders, changes on the old ones, or important variations on the demand market.

# 4 Data Monitoring Module and Data Bases

The data monitoring module is the one in charge of relating the information, given by the user, with the databases that the system has access to obtain the information needed to carry on the corresponding calculations (Figure 4.1). The information in the databases should be constantly updated to reflect reality in order to get reliable results.

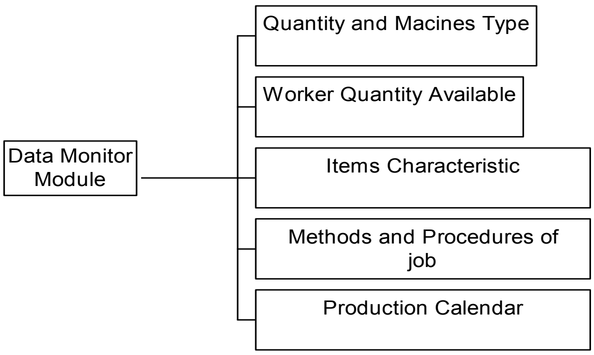



Fig. 4.1 Structure of the Data Monitoring Module (databases)

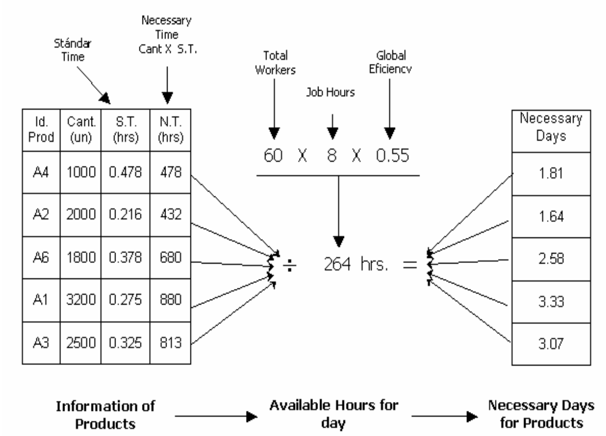



Fig. 5.1 Process to calculate the production time needed per product (in days)

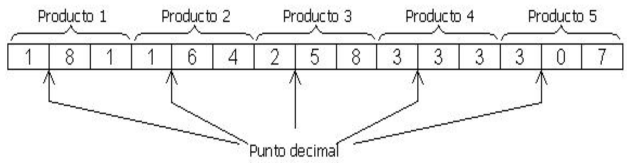



Fig. 5.2 Chromosome representation of the production time per product

# 5 Optimization Module

The optimization module is where the maximum production capacity will be calculated, obtaining as a result the maximum possible quantities that can be produced of each required product. This information will be needed to fulfill the Production or Sales Plan. The optimum outputs from this module are found through the application of the Fuzzy-Genetic Algorithm described next.

## 5.1 Chromosome Representation

The information in the chromosomes represents the production time, in days, to manufacture each product requested in the Production or Sales Plan. This information is calculated from the data available in the corresponding database, like: Hours per shift, working days per week, workers per production line, among others.

For example: In the case of five different products requested in different quantities, and supposing that the daily working hours are 8, with an overall working efficiency of 55%, a total of 60 workers and an average of 14 available days before delivery, the calculations to obtain the time required to produce each product are shown in Figure 5.1.

In Figure 5.2, an example of the chromosome structure to represent the data previously calculated is show. In this case, a fixed position is supposed for the decimal point of each variable or product [6].

## 5.2 Fuzzy Fitness Function

The fitness function is the objective function to be optimized. A characteristic that this function should have is that it might be able to punish bad solutions, and reward the good ones, allowing, in this way, a faster propagation of the last ones [7].

The fitness function will determine which individual chromosomes, in the present population, are good enough to be reproduced, in order to form a new generation. To perform this task we use of a fuzzy fitness function. The fuzzy model is built in three steps as follows [12]: 1.The input and output variables are identified and an adequate name is attached to each of them. In this case, there is only one input variable corresponding to the total number of days needed to produce each product. There is also only one output variable corresponding to the aptitude of each one of the chromosomes evaluated. This aptitude value has a range between 1 and 100. In our application, the chromosomes with a value closest to 1 are selected. 2. The next step is to establish the subsets of the input and output variables, along with an adequate membership function, which in this case are triangle functions. The subsets for the input variable are: near, past and very past (Figure 5.3). The subsets of the output variable are: high, half and low (Figure 5.4) [11]. The central parameter of the functions in the input variable, are adjusted according to the total of available days the company has to fulfill the production plan. .

Near Past Very Past

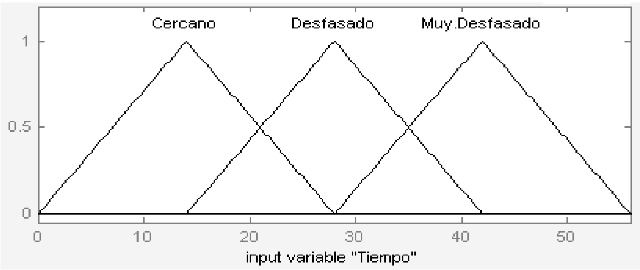



Fig. 5.3 Fuzzy sets of the input variable “Time”

3. Finally, the fuzzy rule-base has to be developed. The fuzzy rules are in the If-Then form and are formulated to establish a mapping from the input variable space to the corresponding output variable space. Usually, the mapping is established according to expert knowledge or by automatically generating the rules from similar examples. The rules generated for our fitness function are the following:
1) If Time is Near Then Aptitude is High.
2) If Time is Past Then Aptitude is Half.
3) If Time is Very Past Then Aptitude is Low.

Low Medium High

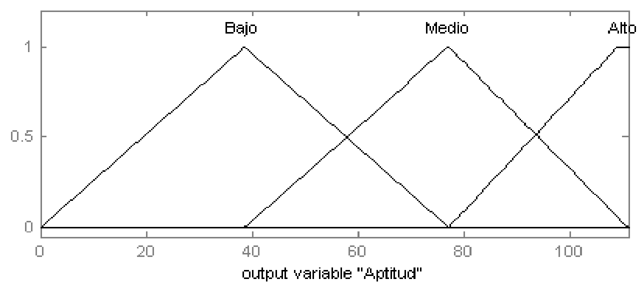



Fig. 5.4 Fuzzy Sets of the Output Variable “Fitness”

## 5.3 Genetic operators

After calculating the aptitude of each chromosome, those with the best fitness values are selected to generate new offspring by the application of the crossover and the mutation operators. With all the generated individuals we get a new population, on which the same process is repeated again until the best solution is found or a stop criterion is reached [9]. To carry out the selection we use the Roulette model, which is a stochastic selection method [6]. For this implementation the following parameters are used: a crossover percentage of 90%, mutation a percentage of 5%, and a total of 200 generations [8].

## 6 Scheduling Module

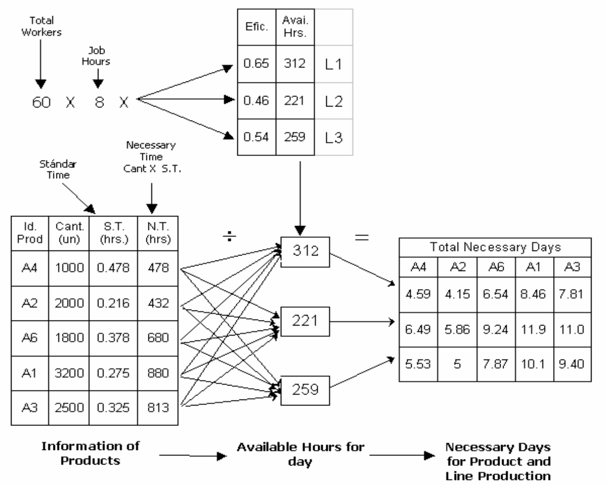
The main function of the scheduling module is the one of carrying out the optimum assignment of products to be produced in the production lines. By the use of an evolutionary algorithm the system is able to obtain the minimum dead time and the minimum production time, for all lines, coming out with the total number of days in which the scheduled production should be completed.

## 6.1 Chromosome Representation

Evolutionary Programming uses integer numbers to code the solutions to the problem at hand. For this reason the phenotype evolves instead of the genotype, as in the genetic algorithms [6]. The information we want to represent in each chromosome is the amount of days needed by production line to manufacture each requested product, under normal conditions. The normal conditions are those that may be established from the information stored in the related data bases. Figure 6.1 shows an example of the structure of the chromosomes to represent the refered data.

## 6.2 Fitness Function

The total sum of the needed days by each of the production lines is calculated by the fitness function. It chooses the greatest of the totals and executes a difference between this and each of the sums [7]. In parallel, another difference is carried out between the available days and the total sum of days needed at each production line



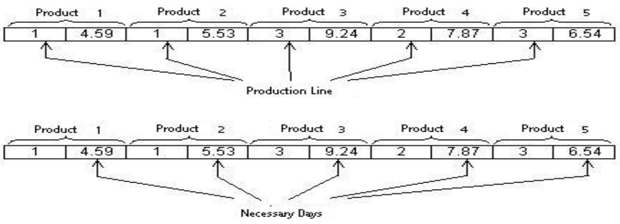



Fig.6.1 Process to calculate the information represented by the chromosomes

Two parallel data defining the fitness of a chain are managed. The first of them is to determine that all the production lines might finish, as much as possible, at the same time, and the second one is to establish the smallest possible time to produce all the products in all the production lines, which is indeed the minimum delay time.

### 6.3 Genetic operators

As in the genetic algorithm, the Roulette selection method is used [9][6]. For this implementation, we use a mutation percentage of 95% [10], and the evolutionary program is run during 150 generations, when most of the chromosomes are almost the same [8].

## 7 System Implementation

All the algorithms designed as well as the computer interface, were created in Matlab version 6.0. This platform was used by the following reasons: it provides an integrated environment which allows the implementation of optimizing functions along with graphical tools to design practical interfaces; it is also very portable and efficient for numeric calculations.

## 8 System Outputs

The results obtained when the genetic algorithm is executed with the data in the Production or Sales Plan, to determine if the company has the enough capacity to fulfill the requested order, are shown in the Figure 8.1.

The results obtained after the process of the evolutionary program are shown in Figure 8.2. At the same time, the resulting data are shown graphically to have a better understanding, in a graph of days per production line. As can be appreciated in the graph in Figure 8.2, the production lines approximately finish at the same time, and the production takes a maximum of 12.9 days, with what the optimization of the assignment is completed. This graph also shows the time at which each production line should finish one product and at which it should begin the manufacturing of another one. This information is of great utility too for the people at the Department of Maintenance, because in this way, they can get a plan ready in advance to carry out the required adjustments in each line to start the production of the next product, reducing in this way the set up time.

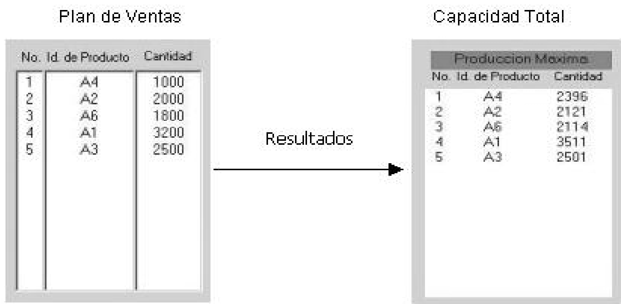



Fig. 8.1 Maximum Production Capacity

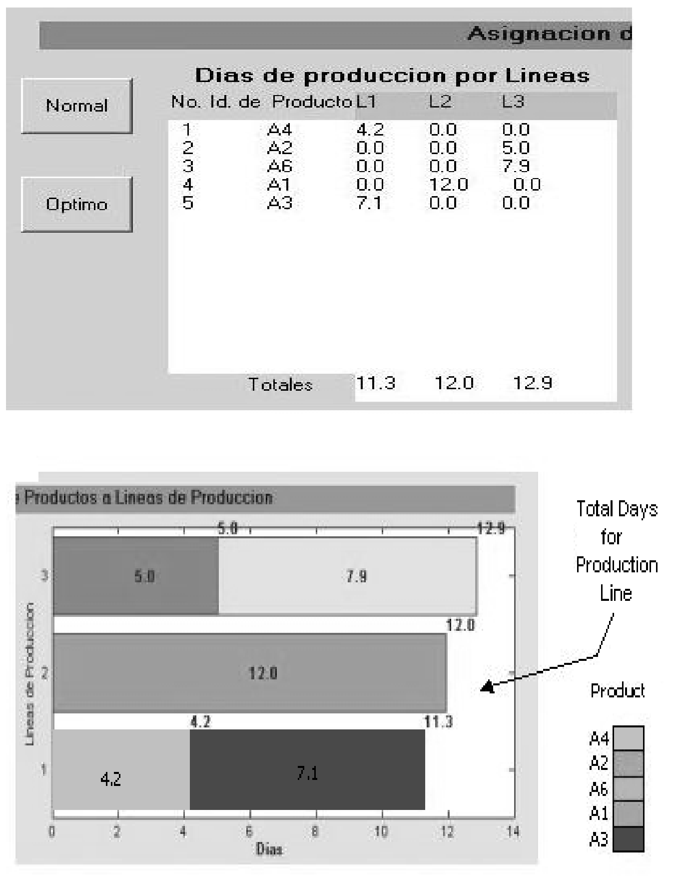



Fig. 8.2 Assignment of products to the production lines.

## 9 Conclusions

The results that can be obtained with the use of this Hybrid System for planning and scheduling optimization, are relevant, because they fulfill the main objectives of determining the maximum production capacity, and the optimization of the production lines usage. Besides, our developed system allows the dynamical change of the production schedule, as required by any sudden modification in sales or market demand. It might be also of great help for companies in the task of making decisions over future situations, like preventing delays and cost changes.

## 10 Acknowledgment

This article is part of a master thesis in Computer Sciences, completed after being supported by a scholarship granted by COSNET, Mexico.

## References

[1] Chiavenato dalberto, "*INICIACIÓN A LA PLANEACION Y EL CONTROL DE LA PRODUCCIÓN"*, Edit. Mc-Graw-Hill, 1993.

[2] Noori Hamid, Radford Russell, "*ADMINISTRACION DE OPERACIONES Y PRODUCCION*", Edit. Mc-Graw-Hill, 1998.

[3] Aguirre Hugo, *PRESUPUESTOS, Actividades de planificación de manufactura (MRP),* Unamosapuntes. http://orbita.starmedia.com/~unamosapuntes/presupuestos/presupuestosonline2.htm

[4] Domínguez Machuca J. A. y Otros, "*DIRECCION DE OPERACIONES". Aspectos Tácticos y Operativos en la Producción y los servicios*, .Edit. Mc-Graw-Hill, 1995.

[5] De la Herrán-Gascón Manuel, *COMPUTACIÓN EVOLUTIVA, h*ttp://www.psycologia.com/articulos/ar-h_gascon01_2.htm/

[6] Coello Carlos A., INTRODUCCION A LA COMPUTACION EVOLUTIVA. LANIA. http://veracruz.lania.mx/~ccoello/

[7] Bentley Peter J., EVOLUTION DESIGN BY COMPUTERS, Edit. Morgan Kaufman Publishers, inc.

[8] Heitkotter Jorg and Beasley David, *GUIDE TO EVOLUTIONARY COMPUTATION,* http://www.etsimo.uniovi.es/ftp/pub/EC/FAQ/www/top.htm, January 2000.

[9] Goldberg David, "*GENETIC ALGORITHMS*" *IN SEARCH, OPTIMIZATION AND MACHINE LEARNING*. Edit. Springer, 1989

[10] Michalewicz Zbigniew, "*GENETIC ALGORITHMS + DATA STRUCTURES = EVOLUTON PROGRAMS".* 3rd. edition, Springer-Verlag, 1993

[11] Ross Timothy J., "FUZZY LOGIC WITH ENGINEERING APPLICATIONS", Edit. Mc-Graw-Hill, 1997.

[12] Reyes García, C. A., SOFT COMPUTING, in McGraw Hill Yearbook of Science and Technology 2002, Edit. Mc-Graw-Hill, 2002, pp 308-310.