Journal of Machine Engineering, Vol. 9, No. 1, 2009

Immune Algorithm,
multi objective optimisation,
fuzzy system.

Iwona WOSIK[1]
Bożena SKOŁUD[1]

TUNING OF A FUZZY SYSTEM FOR CONTROLLING SEARCHING PROCESS IN MULTI OBJECTIVE SCHEDULING IMMUNE ALGORITHM

In the paper the Multi Objective Immune Algorithm (MOIA) for an open job shop scheduling problem (OJSP) is proposed. The OJSP belongs to most both time consuming and most complicated problems in scope of searching space. In the paper schedules are evaluated by using three criteria: makespan, flowtime and total tardiness. MOIA proposes a schedule, which is best one, selected from a set of achieved solutions. An affinity threshold is a parameter that controls equilibrium between searching space and solutions diversity in MOIA. The affinity threshold is defined by using fuzzy logic system. In the paper fuzzy system is tuned by selecting shape, size of fuzzy sets, and fuzzy decisions of an affinity threshold. If the fuzzy system is used then neither the knowledge about the affinity threshold nor influence over searching processes is not required from a decision-maker. The application of the fuzzy system makes the process of decision-making user friendly. In the paper efficiency of MOIA before and after the fuzzy system tuning is compared and computational results are presented.

1. INTRODUCTION

An open job-shop scheduling problem (OJSP) is well-known hard combinatorial optimization problem [3,4,5,6]. Production scheduling problems deal with generating not deleted jobs, minimal production costs. Multi-objective scheduling problems deal with searching a complex space and optimization of the multiple, contradicting objectives. In considered OJSP, schedules are evaluated by using three criteria: makespan, total flowtime and total tardiness. MOIA reaches a schedule selected from a set of obtained solutions using Weighted Aggregation Method (WAM).

Making critical review within the scope of scheduling problems, metaheuristic algorithms have received most interest. Meta-heuristics are applied for the problems with great success when implementation of both problem-specific algorithms and heuristics are not efficient. Heuristic algorithms are less time consuming than meta-heuristic ones. Meta-

[1] Institute of Engineering Processes Automation and Integrated Manufacturing Systems
University of Technology, Poland

heuristic algorithms operate simultaneously on many searching points in a searching space and the obtained solution is transformed into new one. Heuristic algorithms generate a single solution and verify it.

Increasing demand for obtaining better and better schedules causes that the following question appears: which algorithm is more efficient either heuristic or meta-heuristic. The goal is to find an algorithm which gives better solutions within predefined period of time. From some time now, IAs have aroused increasing interest. IAs as well as Genetic Algorithms (GAs) belong to meta-heuristic methods and both are based on biological immune principles. GAs have many common features such as implicit parallelism embedded in the evolutionary mechanism and the hereditary-like process.

The literature review follows, GAs have two shortcomings: premature convergence of solutions to the local optima and lack of the local search ability. Since GAs have not got the performance of genetic search over the subspace of a local optima many methods of hybridization have been proposed to solve the problems. Rekiek proposes a grouping genetic algorithm (GGA) where specific encoding scheme is used, so genes in chromosomes are represented by relevant structures of grouping problems. GGA uses special genetic operators to suit the new encoding scheme. Those aspects avoid the weakness of GAs [9]. Arroyo proposes a multi-objective local search (MOLS) procedure to improve solutions reached by GA. MOLS preserves dispersion in the population and searches simultaneously multi-objective local space [1]. Liaw presents a hybrid genetic algorithm (HGA) with a local improvement procedure as an add-on extra loop [4]. Murata proposes a multi-objective genetic algorithm with two characteristic features to overcome premature convergence: first one is a selection procedure, second one is an elite preserve strategy [7]. A hybrid genetic algorithm (HGA) applying local search procedure is proposed by Ponnambalam [8].

Discussing critical review within the scope of IAs, Liao conclude that IAs are fruitful searching algorithms, because of: (a) ability of IA to adjust automatically to produce correct quality of solutions, (b) maintenance of high diversity of solutions in the searching space, (c) ability of IA to make solutions avoiding to premature convergence [5]. Liao compares IA with standard GA and proves superiority of IA. Zandieh compares IA with a Random Key Genetic Algorithm (RKGA) [14]. Results confirm that IA outperforms RKGA.

IA presented in this paper was compared with GA, and the same conclusion was stated [10]. In the paper, for OJSP the Multi Objective Immune Algorithm (MOIA) is proposed. An affinity threshold is a parameter that controls equilibrium between searching space and solutions diversity in MOIA. The affinity threshold is defined by using fuzzy logic system (FS). In the paper, FS is tuned, by selecting shape, size of membership functions, and fuzzy decisions of the affinity threshold. If the fuzzy system is used the knowledge about the affinity threshold and their influence over searching processes is not required from a decision-maker. The application of FS in MOIA makes the process of decision-making user friendly. The efficiency of MOIA before and after the fuzzy system tuning is compared and computational results are presented.

2. PROBLEM FORMULATION

The OJSP is a scheduling problem described by *I* jobs, $i = (1,2,...,I)$ and *J* machines. $j = (1,2,...,J)$. Each job consists of *J* operations, and each operation requires a different machine. A workflow is not unidirectional. Each machine can be an input and output of the workflow. Each operation is characterized by processing time. There are no precedence relations between operations (operations order is arbitrary). Machines are available continuously. Following constrains have to be considered:

1. Non-preemptive constrain - at any time, at most one operation of each job can be executed. Let c_{ij} describes the completion time of job *i* on machine *j* and t_{ij} denotes the processing time of job *i* on machine *j*. For the job *i*, if operation on machine *k* precedes operation on machine *j*, the following constraint has be fulfilled:

$$c_{ij} - t_{ij} \geq c_{ik} \tag{1}$$

2. Non-reentrant constrain – at any time, at most one operation can be executed on each machine. Let us consider two jobs *i* and *l*, both of them have to be executed on the same machine *k*. If the job *l* precedes the job *i*, the following constraint has to be fulfilled:

$$c_{ik} - c_{lk} \geq t_{ik} \tag{2}$$

Production systems deal with multiple contradictive objectives. Usually, optimising one objective function leads to deterioration others therefore a Pareto optimal solution is searched. The goal of the MOIA is to generate a Pareto curve that gives information about trade-offs between contradictive objective functions. The decision-maker can use two, three or four optimization criteria from the following: makespan minimization $C_{\max} \to \min$, total tardiness minimization $T \to \min$, total flowtime minimization $F \to \min$, total idle time of machines minimization $I \to \min$.

$$f(x) = \sum_{o=1}^{O} fs_o(x) \cdot w_o \tag{3}$$

$$fs_o(x) = f_o(x) / f_o(x^*) \tag{4}$$

$$fs_o(x) \leq fs_0(x^*) \tag{5}$$

where: *f(x)* = scalar fitness function; $fs_o(x)$ = scalar criterion *o*; $f_o(x^*)$ = maximal value of criterion *k*; w_o = priority of criterion *o, o =1,2…,O.*

In order to evaluate schedules the Weighted Aggregation Method (WAM) is applied. In WAM, weighted objective functions are combined into a scalar fitness function (3). Weights indicate priority of criteria. If weights assigned to objective functions are constant, searching direction is also constant in the multi dimensional search space. If weights are randomly assigned to objective functions, the direction of search is not constant. The decision-maker using MOIA can define priorities of criteria (constant weights) or let the algorithm select weights at random. MOIA proposes the best schedule from the Pareto optimal solution set.

The purpose is to select best schedule from the Pareto optimal solution set, with minimal scalar fitness function. Another goal of the paper is to integrate MOIA with FS to maintain equilibrium between diversity and quality of solutions. In the paper FS system is investigated in range of efficiency of the MOIA in reaching best schedule for the multi criteria OJSP.

3. TEST PROBLEM

The OJSP is described by a matrix of processes $MP[i \times j]$ (6), a matrix of operation times $MT[i \times j]$ (7), a matrix of production orders due dates $MD[i]$ (8). *i* denotes *i-th* process, $i = (1,2,...,I)$, and *j* represents machine the operation must be executed on, $j = (1,2,...,J)$. Let us consider 1st row of *MP* , 1st process consists of six operations. This 1st operation of 1st process must be processed on 4th machine, 2nd on 1st machine and 3rd operation on 3rd machine, and so on. There are no precedence relations between operations in the OJSP.

$$MP = \begin{bmatrix} 2 & 5 & 3 & 1 & 4 & 6 \\ 1 & 6 & 4 & 2 & 3 & 5 \\ 3 & 2 & 1 & 6 & 4 & 5 \\ 3 & 2 & 6 & 1 & 5 & 4 \\ 2 & 6 & 1 & 5 & 4 & 3 \\ 1 & 2 & 3 & 6 & 4 & 5 \\ 5 & 4 & 3 & 6 & 2 & 1 \\ 3 & 2 & 1 & 6 & 4 & 5 \\ 3 & 2 & 6 & 1 & 5 & 4 \\ 6 & 3 & 4 & 5 & 2 & 1 \\ 6 & 1 & 4 & 2 & 3 & 5 \\ 3 & 2 & 1 & 6 & 4 & 5 \end{bmatrix} \qquad MT = \begin{bmatrix} 6 & 2 & 3 & 2 & 7 & 1 \\ 5 & 6 & 6 & 3 & 2 & 6 \\ 1 & 10 & 2 & 3 & 2 & 6 \\ 5 & 1 & 4 & 1 & 4 & 1 \\ 3 & 8 & 4 & 1 & 2 & 2 \\ 3 & 1 & 3 & 2 & 7 & 5 \\ 1 & 2 & 1 & 7 & 3 & 3 \\ 1 & 3 & 6 & 2 & 4 & 2 \\ 5 & 3 & 5 & 2 & 6 & 2 \\ 2 & 1 & 2 & 4 & 4 & 3 \\ 5 & 5 & 5 & 6 & 6 & 6 \\ 6 & 1 & 4 & 2 & 3 & 5 \end{bmatrix} \tag{6,7}$$

$$MD = \lfloor 75_, 80_, 900_, 70_, 100_, 50_, 65_, 40_, 85_, 55_, 85_, 100 \rfloor \tag{8}$$

$$V = \lfloor 0.4_, 0.2_, 0.4 \rfloor \tag{9}$$

The MT (7) describes operation times. Duration time of 1st operation of 1st process equals 2, duration time of 2nd operation of 1st process equals 6, 3rd operation of 1st process occupies machine through 3 time units. The due date of production process *i* is determined and described in MD (8). The due date of 1st process equals 75, and of 2nd process equals 80.

The goal is to find a best compromised schedule of 12 processes processed on 6 machines, which is the most compromise solution for three objective functions: $C_{\max} \to \min$, $F \to \min$ and $T \to \min$. Decision maker defines priority of criteria to evaluate schedules (20). Priorities (weights) of 1st and 3rd criteria equal 0.4, the priority of 2nd criterion equals 0.2 (9).

4. THE IMMUNE ALGORITHM APPLICATION FOR SOLVING SCHEDULING PROBLEM

For solving scheduling problems the MOIA (ang.: Multi Objective Immune Algorithm) software has been elaborated. MOIA consists of following modules: data interface, coding, multi criteria optimisation.

Two stages are distinguished in proposed MOIA. First one mimics an exogenous activation of the immune system and second one corresponds to an endogenous activation. The exogenous activation is stimulated by a temperature parameter. A pathogen presence fuels defending reaction of the immune system. This phenomenon is used in the MOIA in order to find a promising search direction (a vector of criteria weights) in the solutions space. The endogenous activation is stimulated by the pathogen presence. In the endogenous activation, antibodies recognizing the pathogen produce offspring in the process of clonally proliferation. Offspring mutate to better match to the pathogen. The pathogen represents scalar fitness function (3), and antibodies correspond to solutions of the problem. Diversity of produced offspring is controlled in processes of stimulation (similar antibodies activation) and suppression (similar antibodies elimination). First immune response is carried out if the exogenous and endogenous activations are stimulated by pathogen attacking the body for the first time. Some of antibodies well-matched to the pathogen survive and are transformed into memory cells for more effective destroying the pathogen in a secondary immune response.

Input parameters are following: number of objective functions $|O|$, primary size of the initial population $|N|$, number of iterations $|M|$ in the endogenous population, number of iterations $|P|$ in the exogenous population (the exogenous activation starts from Step 3 to 9), number of admissible identical solution $|S|$ in a neighborhood $N(p_k)$ of antibody p_k, affinity threshold *Td*, temperature parameter *Tt*.

The exogenous activation

Initial antibodies are randomly generated on a feasible solutions space. Chromosome is represented by a permutation of operations.

As an example, let us consider the OJSP described by J=2 and I=3. The antibody codes the schedule described by (10). The process of decoding starts from 1st operation of 1st job, with the highest priority, which has to be processed on 1st machine. After 2nd operation of 1st job on 2nd machine, the 1st operation of 3rd process is scheduled. After 2nd operation of 3rd job, 2nd operation of 2nd process is scheduled.

$$p = \begin{bmatrix} (1.1.1) & (2.1.2) \\ (1.3.1) & (2.3.2) \\ (2.2.1) & (1.2.2) \end{bmatrix} \tag{10}$$

The process of antibody creation starts from random selection of gene from DNA Library and is continued up to the chromosome completion. Genes stored in DNA Library represent operations described by number of operations, number of jobs, number of machines, for example $(1.2.3)$. A sub-chromosome represents a chain of operations of job (has to be processed). The initial population size equals $K = |O| \times |N|$. Dispatching rules are used to generate few schedules for increasing probability of finding interesting points in the multi-criteria searching space [12].

The best vector of criteria weights is searched in multicriteria space, in Step 3. The promising search direction is the best vector of weights of criteria used for evaluating the schedule. The vector of weights of criteria (gives minimal value of (11)) can be selected at random. Sum of weights must be equal to 1. If $|M| \geq 1$, average scalar function of generation *m* (11) is computed. Condition (12) is verified each generation of the endogenous activation. The promising vector of weight is memorized if (12) is satisfied. The advantage of searching for the best vector of objective weights is reduction of computational time. Step 3 can be omitted, if the decision-maker defines priority of criteria (constant vector of weight).

$$\overline{F^{m}} = \frac{\sum_{k=1}^{K} F(p_k)^m}{K} \tag{11}$$

$$\overline{F}^{m} < \overline{F}^{\min} - Tt \tag{12}$$

where: p_k = antibody *k*, $k=(1,2,\ldots,K)$; $F(p_k)^m$ = scalar fitness function of antibody *k* in generation *m*; $\overline{F}^{m}$ = average scalar fitness function in generation *m*; $\overline{F}^{\min}$ = minimal reached average scalar fitness function; Tt = temperature parameter.

The endogenous activation

The initial population is divided into sub-populations. Number of sub-populations equals $|O|$, $o = \{1,2,\ldots,|O|\}$. Antibodies are decoded into schedules in each sub-population and evolved according to the single criteria: $C_{\max} \to \min$ for $o=1$, $T \to \min$ for $o=2$, $F \to \min$ for $o=3$, $I \to \min$ for $|O|$. Antibodies are selected to create matting pool at the beginning of evolution process of antibodies in sub-populations. Parents are randomly matched in couples. In the sub-populations $1 \ldots |O|$, "position-based operation crossover" is applied. Next, the elite selection procedure is carried out. Better individuals undergo "insertion mutation" and also the elite selection procedure takes place. The probability of crossover or mutation has been assumed as 1, this means that all chromosomes undergo evolution.

Schedules are evaluated using (3) in the sub-population $|O+1|$. A hipermutation operator is used for generating new offspring in the sub-population $|O+1|$. Adjacent genes of one parent's string exchange in the hipermutation operation. Hipermutation rate *h* of antibody k depends on $F(p_k)^m$ (13). Best antibodies do not mutate, worst antibodies mutate intensively. Elite selection procedure is carried out after the process of antibodies' evolution in the sub-population.

$$h=\begin{cases} 0 \;\; if \;\; F(p_k)^m \in \left[F(p_c)^{m,\min}, \left(F(p_l)^{m,\min}+\Delta_F\right)\right) \\ 1 \;\; if \;\; F(p_k)^m \in \left[\left(F(p_c)^{m,\min}+\Delta_F\right), \left(F(p_l)^{m,\max}-\Delta_F\right)\right) \\ 2 \;\; if \;\; F(p_k)^m \in \left[\left(F(p_l)^{m,\max}-\Delta_F\right), F(p_l)^{m,\max}\right] \end{cases} \tag{13}$$

$$\Delta_F=\left(F(p_l)^{m,\max}-F(p_c)^{m,\min}\right)/3 \tag{14}$$

where: p_k , p_l , p_c = antibodies, $F(p_c)^{m,min}$ = minimal scalar fitness function of antibody c in generation m; $F(p_l)^{m,min}$ = maximal scalar fitness function of antibody l in generation m; Δ_F = threshold of hipermutation rate.

Similar antibodies are identified after combining sub-populations $1...|O|$ into sub-population R. Let assume that, the antibody p_k belongs to a neighborhood $N(p_k)$. The antibody p_{k+1} belongs to the $N(p_k)$ if a degree of stimulation between the p_k and p_{k+1} is smaller than or equal to Td. The degree of stimulation is counted using the Hamming distance. The higher the stimulation threshold is the smaller Hamming distance is.

Next, the neighborhood reduction is done. Number of similar antibodies belonging to the $N(p_k)$ has to be $<|S|$. The process of reduction of identical schedules (identical solution are removed form the neighborhood $N(p_k)$) helps to maintain diversity of solutions.

Elite selection strategy is carried out between two sub-populations: R and $|O+1|$. Removed schedules from population R are replaced with ones from the sub-population $|O+1|$.

The best solution is copied to the immune memory in each generation. During secondary immune response, neighboring solutions of schedules contained in the immune memory are searched in order to find locally better solution [12]. The superior solution from the memorized solutions is an optimal or near to Pareto optimal solution for given search direction.

The solution of the scheduling problem should be obtained within number of iterations $|P|$ determined by the decision maker.

5. PARAMETER DEFINING USING FUZZY SYSTEM

An affinity threshold Td is a parameter that controls equilibrium between searching space and solutions diversity in MOIA. In MOIA, Td is defined by using fuzzy logic system (FS).

The FS consists of four elements: a fuzzifier, a fuzzy rule base, an inference engine and a defuzzifier. Controling Td using the FS is presented in Fig. 1. $\overline{F}_m$ represents an average scalar function of two sub-populations R and $|O+1|$. L_m is stated as a number of different schedules in the population m. Td_m represents the affinity threshold in the population m. $\overline{F}_m$ and L_{m1} are input data of the Td fuzzifier. In fuzzification process important control parameters influencing over Td_m are turned into linguistic variable. The triangular function was selected to describe fuzzy sets at the beginning of the process of FS tuning. Membership function with variations in the range $(0, H_{\max})$ of Td_m is presented in Fig. 2. $H_{\max}$ is the Hamming distance between the best and worst schedules. Size of single fuzzy set of $H_{\max}$ is e (where $e = H_{\max}/5$). $\overline{F}_m$ is from the range $(0,1)$. The smaller $\overline{F}_m$ is, the better population quality is (Fig. 3). The range of membership function of L_m depends on a number of different schedules (Fig. 4). Maximal number of possible schedules equals to the size of population $\langle 1, K \rangle$. Size of single fuzzy set of L_m is r (where $r = K/5$). Fuzzy sets are updated in each generation.

Fig. 1. The affinity parameter selected by the FS (based on [2])

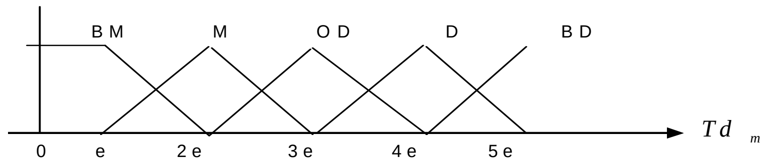

Fig. 2. The membership function of Td_m

Fig. 3. The membership function of $\overline{F}_m$

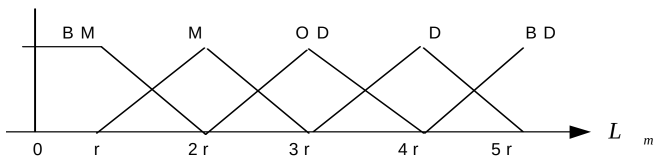

Fig. 4. The membership function of L_m

BM is the abbreviation of “very small”, M – “small”, OD – “suitable”, D – “big”, BD stands for “very big”. Fuzzy decisions for Td_m are made experimentally. The purpose is to maintain equilibrium between quality and diversity of solutions. The example of fuzzy decisions for Td_m is presented in Table 1.

Table 1. The fuzzy decisions of Td_m

$\overline{F}_m$ \ L_m	**BM**	**M**	**OD**	**D**	**BD**
BM	BD	BD	BD	D	D
M	BD	D	OD	OD	OD
OD	D	D	OD	OD	M
D	D	OD	OD	M	M
BD	OD	OD	M	M	BM

According to [2], calculations included in conditions (described in Table 1) are made. The equation (15) is used to count conditions, contains crisp variables:

$$\mu_{Td_m}(x_1, x_2) = \mu_{\overline{F}_m \cap L_m}(x_1, x_2) = MIN\left[\mu_{\overline{F}_m}(x_1), \mu_{L_m}(x_2)\right] \tag{15}$$

where, x_1 and x_2 = crisp values of FS input data.

Let us consider that $\overline{F}_m$ = 0.6 and L_m = 14 in the population m consisting of K = 20 solutions, H_{max} = 150. The membership function of L_m is presented in Fig. 5.

Fig. 5. The membership function of L_m

Activation levels of conditions are as follows:

$$\text{for } \overline{F}_m = M, L_m = D \rightarrow \mu_{Td_m}(0.6{,}14) = MIN(1{,}0.5) = 0.5 \tag{16}$$

$$\text{for } \overline{F}_m = M, L_m = OD \rightarrow \mu_{Td_m}(0.6{,}14) = MIN(1{,}0.5) = 0.5 \tag{17}$$

According to [2] fuzzy sets are replaced with a singleton (one element set) (Fig. 6), achieved value is counted using (15). Proposed method of defuzzification is sensitive for every change of value of input data. The value, which controls the process of balancing both schedules quality and searching space is counted according to (18). For considered example $Td_m^* = 90$ (19).

$$Td_m^* = \sum_{z=1}^{Z} y_i \cdot \mu_{ci} \tag{18}$$

$$Td_m^* = (90 \cdot 0.5) + (90 \cdot 0.5) = 90 \tag{19}$$

where, Td_m^* = crisp value of output data of FS and μ_{ci} = activation levels of singleton i by the rule and y_i = value of input variable and z = number of rules.

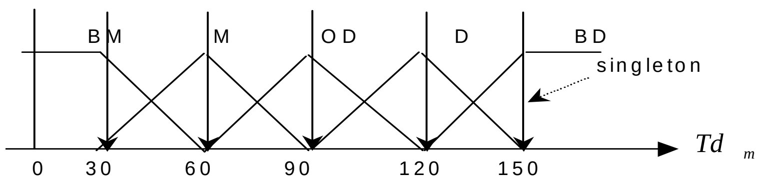

Fig. 6. The membership function of Td_m

Let us consider the OJSP described in the Section 3. There are 12 processes and 6 machines in the OJSP. We estimate the membership function of $\overline{F}_m$. For K = 36 and $H_{\max}$ = 139, input data of FS are as follows: $\overline{F}_m$ = 0.68, L_m = 36. Activation levels of conditions are (20,21).

$$\text{for } \overline{F}_m = BM, L_m = BD \rightarrow \mu_{Td_m}(0.68{,}36) = MIN(0.39{,}1) = 0.39 \tag{20}$$

$$\text{for } \overline{F}_m = M, L_m = BD \rightarrow \mu_{Td_m}(0.68{,}36) = MIN(0.60{,}1) = 0.6 \tag{21}$$

$$Td_m^* = (108 \cdot 0.39) + (81 \cdot 0.6) = 91 \tag{22}$$

Following problem has appeared: the value of Td_m^* = 91 (22) given by FS does not make equilibrium between searching space and quality of reached solutions. The Hamming distances between solutions are in the range $\langle 120{,}139 \rangle$. If the Hamming distance between the p_k and p_{k+1} is smaller than or equal Td_m^* = 91 the antibody p_{k+1} belongs to the $N(p_k)$. For the Hamming distances between solutions $\langle 120{,}139 \rangle$ all solutions differ, and no one is removed in the process of suppression. Detailed values of Td_m reached by FS in each of 10 generations are presented in Table 4. In Section 6 of the paper, sizes of fuzzy sets, and fuzzy decisions in the inference engine to improve ability of FS for controlling Td_m^* are examined.

6. THE FUZZY SYSTEM TUNING

In the process of FS tuning, the membership function of $\overline{F}_m$ was modified (Fig. 7). FS input date are as follows: $\overline{F}_m = 0.73$, $L_m = 36$. $K = 36$ and $H_{\max} = 147$.

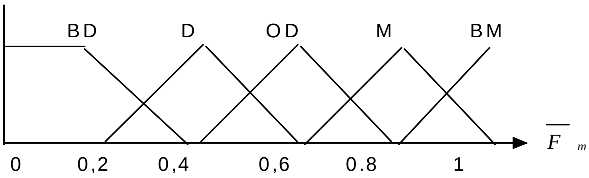

Fig. 7. Shape tuning of the membership function of $\overline{F}_m$

The following problem appears: the value of $Td_m^*=77$ does not make equilibrium between diversity and quality of reached solutions. The Hamming distances between solutions are in the range $\langle 116,138 \rangle$. No one solution is removed in the process of suppression. Detailed values of Td_m reached by FS in each of 10 generations are presented in Table 4.

During simulations it has been noticed that $\overline{F}_m$ oscillates in the range $(0.6,0.7)$, so in this range the membership function of $\overline{F}_m$ was modified (Fig. 8). Simulations were made again. In first simulation, the Hamming distances between obtained solutions are in the range $\langle 118,139 \rangle$. Output data of FS (Td_m^*) receives to small values. Detailed values of Td_m reached by FS in each of 10 generations are presented in Table 4.

Fig. 8.: Size tuning of the membership function of $\overline{F}_m$

FS input data are: $\overline{F}_m$ in the range $(0.65,0.75)$, and L_m - 36. In fuzzification process, parameter $\overline{F}_m$ often receives linguistic variables BM, M, OD and parameter L_m - BD. The interface engine of Td_m should be more sensitive to input data of FS in order to improve the searching process. Fuzzy decisions for Td_m are modified (Table 2). Simulations were carried out. $H_{\max}$ receives value in range $\langle 116,134 \rangle$. Detailed values of Td_m given by FS after the inference engine modification are presented in Table 4. FS does not propose value of Td_m

within range $\langle 116{,}134 \rangle$. $Td_m < 102$ means no one solution belongs to one neighborhood. The following modification of the inference engine of Td_m are done in order to make the inference engine of Td_m even more sensitive to input data of FS (Tab.3).

Table 2 The fuzzy decisions for Td_m - first modification

$\overline{F}_m$ \ L_m	**BM**	**M**	**OD**	**D**	**BD**
BM	BD	BD	BD	BD	BD
M	BD	BD	D	D	D
OD	BD	D	D	OD	OD
D	BD	D	OD	OD	M
BD	BD	D	OD	M	BM

Table 3. The fuzzy decisions for Td_m - second modification

$\overline{F}_m$ \ L_m	**BM**	**M**	**OD**	**D**	**BD**
BM	BD	BD	BD	BD	BD
M	BD	BD	D	D	D
OD	BD	D	D	D	D
D	BD	D	OD	OD	M
BD	BD	D	OD	M	BM

Table 4. Input data $\overline{F}_m$ and autput data Td_m of FSII

		No. of generation										Values of criteria of best schedules		
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	$C_{\max}$	F	T
FSII before tuning	Td_m	96	95	91	90	91	90	87	92	95	88	55	521	8
	$\overline{F}_m$	0,69	0,68	0,67	0,67	0,67	0,67	0,64	0,66	0,65	0,65			
After shape tuning of $\overline{F}_m$	Td_m	77	74	70	72	71	70	70	70	69	67	56	498	3
	$\overline{F}_m$	0,73	0,71	0,70	0,69	0,69	0,68	0,68	0,68	0,68	0,68			
After size tuning of $\overline{F}_m$	Td_m	70	74	68	67	74	65	67	67	64	66	60	593	0
	$\overline{F}_m$	0,66	0,65	0,65	0,65	0,64	0,64	0,64	0,63	0,62	0,61			
First modification of the fuzzy decision of Td_m	Td_m	100	102	99	96	94	93	92	92	94	90	55	513	3
	$\overline{F}_m$	0,68	0,67	0,66	0,65	0,64	0,64	0,64	0,64	0,63	0,63			
Second modification of the fuzzy decision of Td_m	Td_m	135	132	132	134	131	125	130	129	128	130	55	543	6
	$\overline{F}_m$	0,67	0,75	0,75	0,76	0,76	0,71	0,76	0,76	0,74	0,75			

Simulations were carried out. Detailed values of Td_m reached by FS after the inference engine modification are presented in Table 4. Td_m receives values in the range $\langle 125{,}135\rangle$. For example in first simulation, Td_m=135, the Hamming distances between solutions are in the range $\langle 119{,}148\rangle$. Solutions with the Hamming distance below 135 are removed in the process of suppression. Solutions with the Hamming distance above 135 do no belong to the same neighborhood. Modification according to Table 3 makes the inference engine more sensitive in range of linguistic variable M and OD for $\overline{F}_m$.

Five stimulations were made of MOIA with Td_m given by either decision maker or by FS (after tuning) to compare the efficiency of MOIA integrated with FS. Best solutions reached by MOIA are specified in Table 5. For input data: $|O|=3$, $|N|=6$, $|M|=8$, $|P|=10$, $|S|=2$ stimulations were made. MOIA reaches two schedules with not deleted jobs for $Td_m = 80$, one schedule for ($Td_m = 120$), three schedules for Td_m defined by FS. MOIA integrated with FS does not reach significant better schedules. MOIA integrated with FS has reached best Pareto optimal solution. Values of criteria are $C_{\max} = 55$, $F = 525$ and $T = 0$ for the best schedule (Fig. 9). Using FS integration of the decision-maker is not required.

Best schedule for OJSP described by (17-20) has been reached by MOIA integrated with FS. Gantt chart of the best schedule is presented in Fig. 9.

Table 5. Values of criteria of schedules reached by MOIA with Td_m defined by the decision manner and by FS

MOIA ($Td_m = 80$)			MOIA ($Td_m = 120$)			MOIA after FS tuning		
$C_{\max}$	F	T	$C_{\max}$	F	T	$C_{\max}$	F	T
56	**528**	**0**	58	527	6	55	543	6
59	551	0	60	525	2	61	492	0
58	514	2	**56**	**529**	**0**	56	547	2
59	523	15	58	536	4	62	521	0
57	534	9	56	565	1	**55**	**525**	**0**

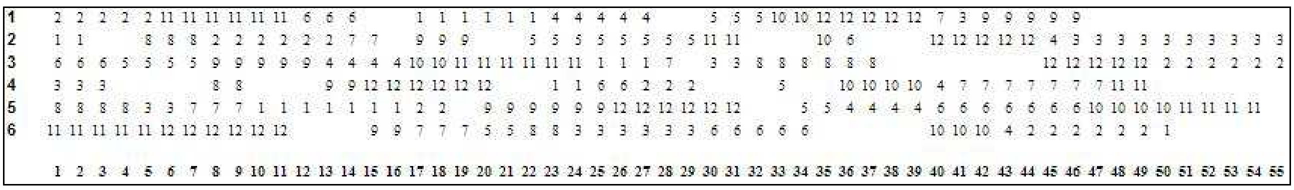

Fig. 9. The Gantt chart of the best schedule reached by MOIA after FS tuning

7. SUMMARY

MOIA is multi criteria optimization aiding system for scheduling problem. In MOIA, the decision maker can use two, three or four criteria to evaluate schedules. The Open Job

Shop Problem (OJSP) of twelve jobs and six machines is presented. For evaluating schedules three criteria were taken into account: the makespan, total flowtime and the total tardiness minimization. The OJSP is used in researches the efficiency of MOIA as it is more complicated problem than Job Shop Problem.

In MOIA control parameter of searching space can be defined using Fuzzy System (FS). In the paper, FS is tuned and results of computational simulations are presented. MOIA integrated with FS does not improve efficiency of MOIA. Best Pareto optimal schedule has been reached using MOIA integrated with FS. FS make the process of decision-making user friendly.

REFERENCES

[1] ARROYO J, ARMENTANO V., *Genetic local search for multi-objective flowshop scheduling problems*, European Journal of Operational Research, No. 167, 2005, 717-738.

[2] DRIANKOY D. HELLENDOOM H., *Wprowadzenie do sterowania rozmytego*, Wydawnictwa Naukowo-Techniczne, Warszawa 1996.

[3] LIAO G.CH., *Short-term thermal generation scheduling using improved immune algorithm,* Electric Power Systems Research, No. 76,2006, 360-373.

[4] LIAW Ch.F., *A hybrid genetic algorithm for the open shop scheduling problem*, European Journal of Operational Research, No. 124, 2000, 28-42.

[5] LIAW CH.F., *Scheduling preemptive open shops to minimize total tardiness*, European Journal of Operational Research, No.162,2005, 173-183.

[6] MATTFELD D. C., *An efficient genetic algorithm for job shop scheduling with tardiness objectives*, European Journal of Operational Research, No. 155, 2004, 616-630.

[7] MURATA T., ISHIBUCHI H., *Multi-objective genetic algorithm and its applications to flowshop scheduling*, Computers ind. Enging Vol.30, Nr. 4, 1996, 956-968.

[8] PONNAMBALAM S.G., RAMKUMAR V., *A multiobjective genetic algorithm for job shop scheduling*, Production Planning and Control, Vol. 12, No.8, 2001, 764-774.

[9] REKIE B., DE LIT P., *Dealing with user's prererences in hybrid assembly lines design*, Second conference on management and control of production and logistics, 5-8 Jully, 2000, France.

[10] SKOŁUD B., WOSIK I., *Multi-objective genetic and immune algorithms for butch scheduling problem with dependent setups*, Recent developments in artificial intelligence methods, Gliwice 2007, 185-196.

[11] SKOŁUD B., WOSIK I., *The development of IA with local search approach for multi-objective Job shop scheduling problem*. 3rd International Conference Virtual Design and Automation, Innovation in Product and Process Development, 2008, 235-242.

[12] SKOŁUD B., WOSIK I., *Algorytmy immunologiczne w szeregowaniu zadań produkcyjnych*. *Zarządzanie przedsiębiorstwem*, Polskie Towarzystwo Zarządzania Produkcją, No1, 2008, 111-127.

[13] WOSIK I., *An immune approach to the open job shop scheduling problem with linear constraints*. 4th International PhD Conference on Mechanical Engineering – PhD 2006, 111-112.

[14] ZANDIEH M., FATEMI GHOMI S.M.T., *An immune algorithm approach to hybrid flow shops scheduling with sequence-dependent setup times*, Applied Mathematics and Computation No.180, (2006). 111-127.