Review of Management and Economic Engineering
Vol. 20, No. 4(82), December 2021, p. 252 - 258, ISSN (print): 1583-624X, ISSN (online): 2360-2155

THEORETICAL ASPECTS REGARDING ARTIFICIAL INTELLIGENCE ALGORITHMS USED IN SERIAL PRODUCTION PROGRAMMING

Florin LUNGU[1*], Mircea TEODORESCU[2*], Oana Eleonora GLOGOVETAN[3*]

1 *Faculty of Industrial Engineering, Robotics and Production Management, Technical University of Cluj-Napoca, Romania, florin.lungu@mis.utcluj*

2 *Faculty of Industrial Engineering, Robotics and Production Management, Technical University of Cluj-Napoca, Romania, m.teodorescu@yahoo.com*

3 *Faculty of Management, "Bogdan Voda" University of Cluj-Napoca, Romania, oanakun@yahoo.com*

* *Correspondence: m.teodorescu@yahoo.com, florin.lungu@mis.utcluj.ro, oanakun@yahoo.com*

Abstract: Artificial intelligence is a relatively new field of computer science that has evolved rapidly in recent years from the stage of laboratory research to that of the field with various spheres of applications. There is currently a whole set of techniques that allow the construction of programs and artificial intelligence systems, systems considered commercial products, successfully applied in the field of mass production. The purpose of this article is to provide a theoretical picture of the artificial intelligence algorithms used in serial production programming. Current research trends in mass production planning are oriented towards the adoption of hybrid techniques, based on specific algorithms of artificial intelligence, which try to combine the advantages of local search strategies (such as optimization by the ant colony model - Ant Colony Optimization, optimization by the particle set model - Particle Swarm Optimization , Genetic algorithms) with those of global search (such as Simulated annealing), in order to determine those optimal and efficient solutions in the case of mass production scheduling.

Keywords: artificial intelligence, series production scheduling

1 INTRODUCTION

"Production scheduling is considered to be the most complex function of the production process. With this function it is established how a product is implemented in the production process. The choice of technology and technological itinerary together with the definition of the manufacturing process is determined in depending on the characteristics of the workpiece (shapes, dimensions, material, tolerance, roughness, functionality, number of parts, etc.)" (Morar, 2007, p.106). The evolution of series production programming over time is closely linked to a series of artificial intelligence algorithms that have led to the emergence of new generations of commercial products in increasingly shorter periods. In the following we will present a series of artificial intelligence algorithms used in various fields of production but mainly in series production.

2 OPTIMIZATION ALGORITHMS (EVOLUTIONARY)

Computational artificial intelligence (IAC) (Chircu, 2017) is an extremely complex interdisciplinary field, and the main specific techniques are Artificial Neural Networks, Fuzzy Logic and Evolutionary Algorithms. Evolutionary Optimization Algorithms is a subdomain of IAC that uses an iterative process of growing and developing the solution population, which is then processed and used for a random search for an optimal solution. There are a number of metaheuristic optimization algorithms specific to these evolutionary techniques, such as:

A. Neural Networks
B. Genetic Algorithms
C. *Swarm intelligence (Swarm intelligence algorithms)* Particle Swarm Optimization; Ant Colony Optimization (Optimization with ant colonies);
D. Simulated Hardening (SA)

The researchers' attention was mainly focused on minimizing the total time associated with the production process, but also on maximizing the use of machines and minimizing waiting times.

A. NEURONAL NETWORKS - is a technique that was considered the forerunner of current artificial intelligence algorithms. Even a brief description of a natural neuron cannot bypass its character as a computing device with a number of inputs (stimuli) and a single output on an axon, an output that usually exists or does not exist (is zero) and can serve as input for another neuron or, in general, for other neurons within so-called synapses. The inputs of a neuron are combined into a single input type by weighted sum according to certain rules of the actual inputs. There is a sensitivity threshold below which the output of the neuron is zero. Exceeding the threshold produces an output, always the same, so the neuron is basically an element with binary output, on only two levels. In general, the function that connects the output of the neuron to its synthetic input is called the activation function.

The threshold activation function usually associated with natural neurons is illustrated in Figure 1. The non-zero sensitivity threshold is observed and a finite-time transition from one state, the one with zero output, to the other state with non-zero output is suggested. As usual, even in the case of neurons it is not possible to instantly vary a size, its output. In a stratified artificial neural network, information processor neural units are arranged in a sequence of three or more layers of neurons.

The outputs of the neurons in one conveniently weighted layer are inputs for the neurons that belong exclusively to the next layer or are outputs of the network if it is the output layer. The first layer receives the inputs (stimuli) from the environment. The last layer produces the outputs, basically the result of a more or less complex calculation. The inputs of the neurons in the inner layers, hidden and of the last layer, the output layer are linear combinations of the outputs produced by the neurons in the previous layer.

The coefficients of those linear combinations are called weights and have a very important role in the so-called training of a neural network, in a learning process that makes a layered structure of neurons to be adapted to a certain technical or technological purpose. The role of any inner, hidden neural layer is to reformulate and reapply the outputs of the previous layer in order to obtain a more capable representation of separating, classifying the data from the network input. The inner layers allow the attachment of a particular semantics to the network input combinations. (Panaitescu, 2007)

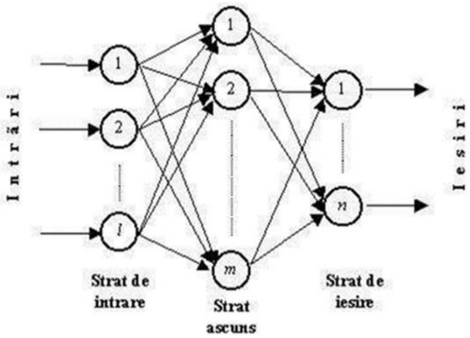

Figure 1. Activation of Natural Neurons (Panaitescu, 2007)

The structure of stratified neural networks can be very different if the number of layers and the number of neurons in each layer are taken into account. Figure 1 shows the structure of a neural network with three layers, one input, one hidden and one output, with l, m, respectively n neural cells. It must be said that the input layer of an artificial neural network usually has only the role of preparing the inputs of the next layer. The neurons in the first layer have a single input that they bring by translation and scaling to appropriate values to be valid stimuli for the neuronal cells in the next layer. Artificial neural networks are already widely used to solve learning problems in various fields. By using existing experimental data, neural networks basically learn the relationships between inputs and outputs.

B. GENETIC ALGORITHMS

- engineering problems with large or very large dimensions can be treated by methods based on genetic algorithms. Establishing the extremes of multimodal functions, optimal structuring and training of neural networks are examples of such problems. Genetic algorithms are a loan from biology and are based on Darwinian evolutionism. It is considered a population made up of individuals described by structures called chromosomes. Chromosomes are usually linear structures, sets of genes. Figure 2 illustrates two individuals by their chromosomes, the genes being represented by colors.

Figure 2. Individual chromosomes (Panaitescu, 2007)

Any population is evolving. The individuals that make it up combine in pairs to generate offspring. The current process is that of cross-combination. The combination results in offspring that are in turn characterized by chromosomes. Their chromosomes result from a cross-reading of the parent chromosomes, broadly according to the diagram in the following figure. At the bottom are represented the specific chromosomes and the resulting offspring. It is not mandatory for two descendants to result from the combination, but in many technical applications the application of the combination operator produces two descendants as in Figure 3. Of course, the switching point of the reading from one chromosome to another can be positioned elsewhere. There may also be several crossing points.

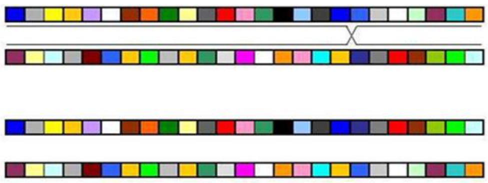

Figure 3. Combination of neurons (Panaitescu, 2007)

In engineering applications we talk about populations of solutions of a problem and the evolutionary determination of the solution of that problem. These are especially complex problems, of excessive dimensionality for which there are no analytical ways of solving, and the enumeration of all acceptable solutions is an illusion. Here, too, as in the case of biological populations, we speak of the better or weaker adequacy of the solutions to the treated problem, just as the individuals of a species are more or less adequate to the problem of survival in an environment generating various challenges. In both cases, the Darwinian principle of natural selection "the most suitable survive" works systematically to adapt solutions to the formulated problem, respectively of individuals to the problem of survival and implicitly of perpetuation.

The procedure is repeated until a given number of successive populations has been generated or a certain adequacy condition has been reached.

Selection - elitist selection is the process by which individuals are selected to participate in the process of reproduction. This involves ordering the individuals of the current population according to the value of the fitness function and selecting the best of them to become parents. The number of individuals who will be chosen to participate in the crossover depends on the value of the crossover parameter.

Crossing - splitting two chromosomes and combining the resulting parts to form two other chromosomes that will be included in the new population is called crossover. The mutation modifies a chromosome in the new population, altering one or more of its genes. The addition of new individuals to the existing population is done taking into account the coding used. In this case, the single-cut crossover method is used.

Mutation - avoiding the blocking of the algorithm in the local optimal points and maintaining the diversity in the population is done with the help of the mutation operator. It involves applying a small change to a gene of the original individual and reintegrating it into the current population. The number of individuals to which the mutation operator will be applied depends on the value of the parameter *pm*.

Note:

- nmax - the maximum number of individuals;
- Gmax - maximum number of generations;

The formalized AG algorithm

1. Initial_population_random_generation ()
2. Assessment of individuals_initial_population ()
3. As long as n ≤ *nmax* and g≤ *Gmax* execute
3.1.Apply_selection_operator ()
3.2.Apply_cross_operator ()
3.3.Apply_mutation_operator ()
3.4.Evaluation_individuals_current_population ()
4. Show_the_ best_individual ()

The advantages of using genetic algorithms refer mainly to the high performance in solving large optimization problems, of great complexity, such as the JSS problem.

The disadvantages of this method can be related to the fact that in the context of such a complex problem, it is difficult to identify the global maximum, establishing appropriate values for specific parameters and defining a fitness function as appropriate for the proposed problem being essential for the correct operation of the algorithm.

C. SWARM INTELLIGENCE. PARTICLE SWARM OPTIMIZATION

Description of the method A population-based stochastic optimization technique whose source of inspiration is social behavior is Particle Swarm Optimization (PSO) and is related to the transmission and sharing of information by living beings such as flocks of birds and shoals of fish. the search process is provided by a set of particles whose movement is characterized by a speed that changes over time depending on the characteristics of the entire system.(Kennedy, Russel 2001).

The system is initialized with a population of pseudo-random candidate solutions and seeks the optimum by updating the population in generations. Individuals, called particles, move through the search space following the current optimal particles and have memory. Thus, each particle has an adaptive speed that directs its flight and stores at each step the best position visited in the search space. (Kennedy, Russel 2001).

At each iteration, the speed of each particle is updated following two values: the best location encountered by that particle in the past and the best location encountered by the flock, according to the following formulas (Bonabeau et.al 1999):

$vid = vid + \emptyset 1 * ()\ (pozid - pid) + \emptyset 2 * rand\ ()\ (pozgd - pid)$

$xid = xid + vid$

where:

Pi = (pid, … ,,) is a particle,

MP = ($P1$, …,) is the set of particles,

Vi = ($Vi1$, …,) represents its velocity,

$Pozi$ = ($pozi1$, …,) the best position visited by particle i,

g is the index of the best individual in the flock,

and Ø1 and Ø2 are constant: the cognitive and social parameters, respectively.

The random values belong to the set {0,1}. A function $fPSO$: $MP \longrightarrow$ [0,1] is also defined, which measures the quality of each proposed solution.

The formalized PSO algorithm

1. Initial_population_random_generation ()
2. Assessment of individuals_initial_population ()
3. How long step <$Imax$ executes
3.1. For each particle P executes
3.1.1. Calculate pBest (P)
3.2. Calculate gBest
3.3. For each particle P executes
3.3.1. Update_particle_speed
3.3.2. Update_particle_position $vid = vid + \emptyset 1 * ()\ (pozid - xid) + \emptyset 2 * ()\ (pozid - xid)$
4. Displays gBest

Advantages of using the PSO method to solve automatic planning problems refers primarily to high performance in solving large, highly complex optimization problems, such as the JSS problem. In addition, the PSO algorithm generally offers much faster solutions compared to other evolutionary algorithms.

The disadvantages of this method can be related to the fact that, in the context of such a complex problem, there is the possibility of blocking in a local optimal solution. Also, establishing adequate values for specific parameters and defining a fitness function as appropriate as possible for the proposed problem are essential for the correct functioning of the algorithm.

3 ANT COLONY OPTIMIZATION

Description of the method. One of the most recent metaheuristics inspired by the natural environment is the optimization based on the study of ant colonies (eng. - "Ant Colony Optimization", ACO) (Blum, 2005).

The identification of a semi-optimal solution is made on the basis of a priori information about its possible structure with additional information about the structure of the previously accepted solution. An ACO algorithm works based on the following principles (Dorigo, Brittari, 2004):

- A set of asynchronous and competing agents, whose formal representation is based on the description of an ant colony, which moves through the states of the problem that correspond to its partial solutions. Agencies move using a set of local, stochastic decisions, which are made based on the values of two parameters: pheromone pathways and attractiveness;
- Through successive movements, an ant builds incrementally the solution of the problem;
- When identifying an acceptable and complete solution, the ant evaluates the solution found and modifies the existing pheromone pathway that will be used in the search for future ants.

ACO algorithms have two important control mechanisms:

- Evaporation of the pheromone pathway decrements all pheromone pathway values to avoid unlimited accumulation of information in the case of components of the solution;
- Centralized actions used to renew global system information.

The computational agent that iteratively builds a solution to a given problem is actually an ant. Intermediate states are nothing but partial solutions. The core of each ACO algorithm is represented by a loop in which, at each iteration, each ant moves (executes a step of the algorithm) from one state to another corresponding to a more complete partial solution (Maneizzo et.al 2004). For an ant, the probability of moving from one state to another depends on the combination of two values:

- The attractiveness that is calculated based on a heuristic from a priori information, representing the general probability of moving to the new state;
- The level of the pheromone pathway that indicates how efficient a certain movement has been in the past, being a current indication of the probability of movement in the new state.

The formalized ACO algorithm

1. Initial_population_random_generation ()
2. Evaluation_of_initial_opults ()
3. As long as the Requirements list ≠ ∅ executes
3.1. For each ant perform
3.1.1. Calculate_the_probability_of_transition (A)
3.1.2. Move (A)
3.1.3. Rate (A)
3.1.4. Pheromone_store ()
4. Show_the_best_ ant ()

The advantages of using the ACO method to solve automatic planning problems mainly refer to the high performance in solving large, highly complex optimization problems, such as the JSS problem.

Disadvantages - the possibility of blocking in a local optimum solution, without going to the global optimum is one of the main disadvantages of this method. Also, setting appropriate values for specific parameters and defining a fitness function as appropriate as possible for the proposed problem are essential for the correct operation of the algorithm.

D. SIMULATED ANNEALING

Description of the method (Van Laarhoven, 1992)

Another suitable method for solving problems of great complexity, especially appreciated for the fact that it manages to avoid the trap of local optimal solutions, managing to move towards global optimal solutions is Simulated Annealing (SA). The notion of hardening, specific to the field of metallurgy, is implemented within the SA as a continuous decrease of the probability of accepting unsatisfactory solutions in the solution search process in order to explore as much as possible from the search space (Michalewicz et al.2004).

Thus, the algorithm starts with a randomly generated initial state to which a certain initial temperature is associated. This temperature gradually decreases as the simulation process progresses until a certain number of steps are performed or the minimum error criterion is met.

The algorithm also involves the use of a fitness function f that measures the quality of the solutions.

The formalized SA algorithm

1. Generate_random_initial_solution ()
2. Initialized_initial_temperature
3. Evaluate_initial_solution ()
4. As long as the current temperature is higher than the minimum running temperature
4.1. Generate_random_intermediate_solution ()
4.2. Rate_intermediate_solution ()
4.3. Compare_intermediate_solution_with_ current_solution ()
4.4. Adjust_temperature
5. Show_final_solution ()

Advantages of using the SA method to solve automatic planning problems refers primarily to high performance in solving large, highly complex optimization problems, such as the JSS problem. Also, this method has a very short running time, quickly reaching a solution.

The disadvantages of this method can be related to the fact that, in the context of such a complex problem, there is the possibility of blocking in a local optimal solution, without moving towards the global optimal. Also, establishing adequate values for specific parameters and how the solution is represented are essential for the correct operation of the algorithm.

The current state of research on the application of AI algorithms takes into account different approaches to artificial intelligence techniques for the field of production planning, which are presented and widely discussed in the literature. Minimizing the total time associated with the production process of waiting times and maximizing the degree of use of machines are aspects that researchers have focused on.

4 CONCLUSIONS

The complexity of the problem of production planning has been a real challenge for researchers around the world, which has led to the attempt to solve this problem through a multitude of methods and techniques that propose to solve it in an optimal way.

A quality planning system must provide the user with a user-friendly interface, be adaptable to performed or the minimum error criterion is met. The algorithm also involves the use of a fitness different manufacturing systems, be fast, optimize the use of available machines, avoid overcrowding of warehouses, etc.

In recent years, automated planning systems have been developed with applicability in academia and industry that have tried to address the complexity of the problem of mass production planning.

These systems are beginning to use solutions based on the use of artificial intelligence algorithms. Some of the techniques specific to Artificial Intelligence (Neural Networks, Particle Swarm Optimization, Ant Colony Optimization, Genetic Algorithms and Simulated Annealing) were presented, each of them being briefly described in the classical form, with its advantages and disadvantages.

The multitude of restrictions that must be met simultaneously in the series production planning process have as an effect the complexity of the production planning problem and have determined the approach of new research directions.

In conclusion, we can say that a possible hybrid method that uses the advantages of several artificial intelligence algorithms could significantly improve the results for mass production.

BIBLIOGRAPHY

Blum, C., (2005), Ant Colony Optimization: Introduction and recent trends, *Physics of Life Review*, vol 2.

Bonabeau, E., Dorigo, M., Theraulaz, T., From (1999) *Natural to artificial Swarm Intelligence*, Oxford University Press, New York;

Chircu F.A., (2017) *Contributions regarding the development of an intelligent system of automatic production planning in flexible manufacturing lines*, doctoral thesis, University of Ploiești Oil and Gas;

Dorigo, M., Brittari, M., (2004), *Ant Colony Optimization, and swarm intelligence*, Springer

Kennedy, J., Russel, E., (2001), *Swarm Intelligence*, Academic Press;

Maneizzo, V., Gamberdella, L., Luigi, F.,(2004), *Ant Colony Optimization, New Optimization Techniques in Engineering*, by Onwubolu, G. C., and B. V. Babu, Springer-Verlag Berlin Heidelberg, 101-117;

Michalewicz, Z, Fogel, DB, (2004) *How to solve it: Modern Heuristics*, Springer;

Morar L., (2007), *Planning and operation of production system*, Ed. FRAUNHOFER IRB VERLAG;

Panaitescu G., M., *Modelarea și Simularea sistemelor de Producție*, Curs pentru învățământul la Distanță, Universitatea de Petrol și Gaze, Ploiești, 2007;

Van Laarhoven, P.J., Aarts, E.H., *Simulated Annealing: Theory and Application*, Kluwer Academic Publishing, 1992.