



**HAL**  
open science

# A Rank Based ACO Approach for Optimal Resource Allocation and Scheduling in FMS Modeled with Labelled Petri Net

Amira Jablaoui, Hichem Kmimech, Layth Sliman, Lotfi Nabli

► **To cite this version:**

Amira Jablaoui, Hichem Kmimech, Layth Sliman, Lotfi Nabli. A Rank Based ACO Approach for Optimal Resource Allocation and Scheduling in FMS Modeled with Labelled Petri Net. 20th International Conference on Intelligent Software Methodologies, Tools, and Techniques (SOMET 2021), Sep 2021, Cancun, Quintana Roo, Mexico. pp.688-699, 10.3233/FAIA210064 . hal-03453230

**HAL Id: hal-03453230**

**<https://hal.science/hal-03453230v1>**

Submitted on 10 Aug 2023

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# A Rank Based ACO Approach for Optimal Resource Allocation and Scheduling in FMS Modeled with Labelled Petri Net

Amira JABLAOUI<sup>a,1</sup>, Hichem KMIMECH<sup>b</sup>, Layth SLIMAN<sup>c,d</sup> and Lotfi NABLI<sup>b</sup>

<sup>a</sup> *University of Monastir, Higher Institute of Computer Science and Mathematics*

<sup>b</sup> *University of Monastir, Research Laboratory of Automatic Signal and Image Processing*

<sup>c</sup> *EFREI Paris, Villejuif, France*

<sup>d</sup> *Université Paris-Saclay, IBISC, 91020, Evry-Courcouronnes, France*

**Abstract.** In this article, we study the NP-Hard combinatorial optimization problem of the minimum initial marking (MIM) computation in labeled Petri net (L-PN) while considering a sequence of labels to minimize the resource consumption in a flexible manufacturing system (FMS), and we propose an approach based on the ant colony optimization (ACO) precisely the extension Rank-based ACO to optimal resource allocation and scheduling in FMS. The ACO meta-heuristic is inspired by the behavior of ants in foraging based on pheromones deposit. The numerical results show that the proposed algorithm obtained much better results than previous studies.

**Keywords.** Labeled Petri Nets, Flexible Manufacturing System, Minimum Initial Marking, Scheduling, Resource Allocation, Label sequence, Ant Colony Optimization;

## 1. INTRODUCTION

A flexible manufacturing system (FMS) is a discrete event system (DES) that is driven by events, more information therefore more and more complex system, which implies the need for powerful tools for modeling these systems [1]. In the literature, Petri nets have been recognized as powerful tools for modeling, analyzing, and evaluating discrete event systems, including flexible production systems, thanks to their graphic and mathematic support [2]. As a mathematical tool, it is possible to set up equations of state, algebraic equations, and other mathematical models governing the behavior of the system. Various traditional optimization techniques are aggregated with different extensions of Petri Nets PN, like fuzzy logic [3], dispatching rules [4], heuristic research [5], [6], [7], [8], and meta-heuristic research [9], [10].

To avoid wasting money and time, optimal scheduling and the allocation of resources are very important for the design of complex systems. Therefore our goal is to reduce the resources allocated by a process (production), and according to the flexibility of the system, we focus on the labeled PN model. To reach this aim, the concrete problem that we are dealing with in this paper is the computation of the minimum initial marking

---

<sup>1</sup>Corresponding Author: Amira JABLAOUI, ISIM Monastir, Tunisia, E-mail: jablaouiamira89@gmail.com.

(MIM) in a labeled PN while considering a sequence of labels (sequence of task/action/event). To find the optimal scheduling, a combinatorial explosion can occur thanks to the number of possible combinations that increase with the length of the label sequence. It's an NP-hard problem [12]. Various methods were proposed in the literature to solve this problem based on different artificial intelligence techniques for the resolution of combinatorial NP-Hard optimization problems. Some articles based on heuristics function, which do not guarantee the optimal solution as in [11], [12]. Others research papers adopted metaheuristics to solve the MIM problem [13], [14], [15], [16], [17]. However, these approaches have different limitations either on quality of solution, or on the complexity of the proposed algorithms.

To reduce the complexity of the MIM computation problem and to obtain an optimal solution in the shortest time, we propose an approach based on the ACO bio-inspired metaheuristic precisely the Rank-Based ACO extension to find better results in a minimum time.

The rest of the paper is structured as follow: In Section II, we briefly introduce the ACO metaheuristic, and present a state of art concerning the MIM computation problem, basic concepts of PNs, and especially the L-PN. In Section III we describe the MIM computation problem in L-PN and its modeling process using ACO algorithms. In section IV, we provide an empirical study compared to previous results to prove the effectiveness of our approach. Finally, section V concludes the paper with hints on future work.

## 2. RELATED WORKS

In this section, we present the previous works dealing with the MIM computation problem in L-PN. Second, we introduce the ACO metaheuristic. Finally, basic concepts of PN and L-PN are presented with a description of the model L-PN that will be used in this paper.

### 2.1. Related Works

Following the recent works that deal with the MIM problem in L-PNs. The authors in [12] propose a recursive algorithm to estimate the value of MIM in L-PN with only observable transitions. They defined two heuristics that make it possible to reduce the complexity of the MIM problem. The drawback of the proposed approach is that the execution takes a long time. The results of MIM with observable transitions in [12] extended to the work [18] and an approach for MIM L-PN with unobservable transitions is proposed. In particular, it assumes that the PNs structure is given and the unobservable transitions in the net are contact-free.

Other works focused on reducing the complexity of the computation algorithm of MIM to obtain a better solution in the shortest time. Therefore heuristics and metaheuristics were adopted to solve the MIM problem in L-PNs. For example, Kmimech et al. [15] adopted the simulated annealing metaheuristic to automatically calculate the MIM on timely manner. The process is to find the transition sequence FS, which makes it possible both to cross the sequence of labels LS (input of the algorithm) and to minimize the number of tokens necessary to process the labels' sequences.

Genetic algorithms GA is a great advantage of less computational time, in [14] the GA is implemented to the MIM estimation problem in L-PNs. M Gam et al. [16] also

proposed an approach based on the genetic algorithm thanks to their potential for the computation of a large number of random sequences and the use of tabu search method to find the MIM of the different random sequences in the L-PNs.

In [13], the authors suggested an approach based on the GRASP metaheuristic to solve and optimize the MIM problem. A Abdellatif et al. [17] also proposed an approach based on GRASP is proposed to automatically compute the value of MIM in a timely fashion.

## 2.2. Ant Colony Optimization

The behavior of virtual ants is inspired by that of real ants, based on the deposit of pheromones in the path. This form of indirect communication, named **stigmergy**, intensifies research around the most promising parts of the research space. In fact, the indirect communication of ants has inspired many researchers and was successfully applied to several optimization algorithms for NP-hard problems, such as the travel salesman problem (TSP) [22], construction site layout problem [19], vehicle routing problem (VRP) [20], Flexible Job Shop Problem (FJSP) [21].

The use of ACO requires choosing a formal representation of the optimization problem to be treated and defining the solution construction process by the ants using this representation [23].

Fundamentally, the ACO algorithm is based on three phases:

- Representation of the artificial pheromone,
- Solution building depends on the amount of the deposited pheromone,
- Pheromone update includes the deposit of pheromone and its evaporation processes.

## 2.3. Basic Concepts of Petri Nets

PNs are modeling tools based on graph theory which makes it possible to model parallel, concurrent and simultaneous activities in scheduling and then solve the problems of blocking, overflows, resources management,...etc.

### 2.3.1. Mathematical Model of PNs

Formally a Place-Transition PN is a quadruplet  $R = \{P, T, PRE, POST\}$

- $P = \{p_1, \dots, p_n\}$  is a finite set of places, denoted graphically by circles.
- $T = \{t_1, \dots, t_m\}$  is a finite set of transitions, denoted graphically by horizontal bars.
- PRE: is the application of the previous places  $P \times T \rightarrow N$ .
- POST: is the application of the following places  $T \times P \rightarrow N$ .

A set of arcs, denoted by arrows that join the places to the transitions and the transitions to places;

By default, the incoming (PRE) and outgoing (POST) arcs are equal to 1 if it is not determined in the model.

We can also add the notion  $C = POST-PRE$  called the incidence matrix.

### 2.3.2. Dynamic Behavior of PNs

To define the state of a system modeled by a PNs, it is necessary to complete the PN by a marking. This marking consists of placing an integer number (positive or null) of marks (or tokens) in each place of the PN.

A marked PN is defined by  $N = \langle R, M \rangle$

R: is a Place-Transition PN.

M: is an initial marking of the PN,  $M(p)$  indicates the number of tokens contained in place  $p$ .

### 2.3.3. Transition Firing Rules

The tokens, which materialize the state of the network at a given moment, can pass from one place to another by firing a transition. A transition  $t$  is said to be sensitized, passable, or pullable:

$$\forall p \in P, M(p) \geq PRE(p, t) \quad (1)$$

If  $t$  is a passable transition with the marking  $M$ , the firing of  $t$  produces a new marking  $M'$ :

$$\forall p \in P, M'[p] = M[p] - PRE(p, t) + POST(t, p) \quad (2)$$

### 2.3.4. Labelled PN (L-PN)

A labeled PN (L-PN) is an extended PN, which is formally defined as a triplet  $LN = (N, L, TL)$

- N is a Marked PN.
- L is a set of labels.
- TL is the transitions labeling function, it assigns a label for each transition  $T \rightarrow L$ .

The marking graph of L-PNs is

$$M = \begin{bmatrix} m_0 \\ \vdots \\ m_i \\ \vdots \\ m_n \end{bmatrix}, \quad m_i (M[p]) \text{ denoted the number of tokens in a place } p.$$

The number of tokens contained in a place reflects the number of its resources.

In an FMS, the optimal scheduling should involve minimal resource consumption without deadlock in the production. With the L-PN model, the notion of label supports the modeling of flexibility in the FMS, where several possible paths to execute the same sequence of tasks (labels in L-PN). To find the optimal resource allocation in the FMS modeled with L-PN, it's to find the firing sequence that minimizes the number of tokens in the IM of their L-PN model while considering a sequence of tasks (labels in the L-PN).

The model L-PN used in this paper (figure 1), is a parallel production chain studied in the literature. It is composed of 10 places and 12 transitions:

Places =  $\{P_1, \dots, P_{10}\}$ , Transitions =  $\{T_1, \dots, T_{12}\}$ ,  $L = \{a, b, c, d, e, f, g, h\}$ . The labeling function TL, where all transitions are observable:  $TL(T_1)=e$ ,  $TL(T_2)=f$ ,  $TL(T_3)=TL(T_5)=a$ ,  $TL(T_4)=TL(T_6)=b$ ,  $TL(T_7)=TL(T_8)=c$ ,  $TL(T_9)=TL(T_{10})=d$ ,  $TL(T_{11})=g$ ,  $TL(T_{12})=h$ .

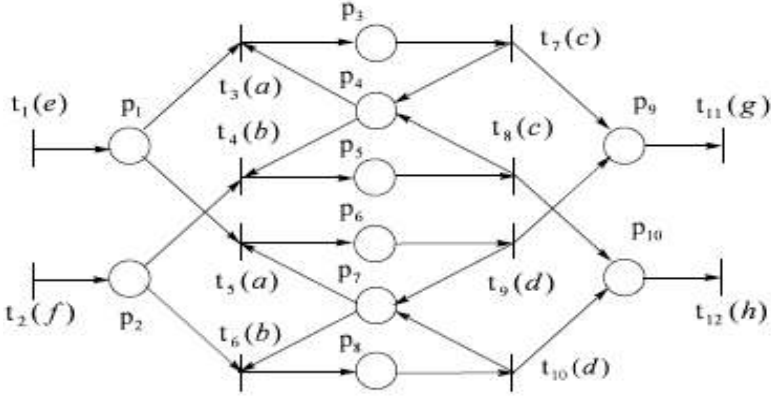


Figure 1. The studied L-PN.

### 2.3.5. Initial Marking Computation of L-PN

The process to compute the MIM consists in finding the firing sequence FS which allows the process of the labels sequence LS, and calculating the marking initial necessary to fire this firing sequence. As noted in [14], the computing of IM is done according to the aforementioned transitions firing rules.

For an L-PN and an LS with known length, the iterative formula to calculate the required IM is:

$$\forall p \in P, M_j[p] = M_{j-1}[p] + PRE(p, FS[j]) - POST(FS[j-1], p) \quad (3)$$

Where  $j$  is an iterator that varies from 1 to  $n$ .  $M_j$  and  $M_{j-1}$  are the computed marking at the iteration  $j$  and  $j-1$ . FS is a line vector of length equal to number of transition of the L-PN model, contain all transitions of the model.

## 3. MINIMUM IM COMPUTATION IN L-PN USING ACO

In this section, first, we formulate the MIM computation problem in L-PN with a brief description of the L-PN model used in our approach, while specifying the fitness function that we search to minimize. Finally, we represent our approach to solve the MIM problem based on the Rank-based ACO algorithm.

### 3.1. Minimum IM Computation Problem in L-PN

As the formulation in previous works, the problem of MIM in L-PN can be defined as a triplet  $LN = (LN, LS, IMC)$

- $LN = (N, L, TL)$  is an L-PN.

- LS is a label sequence defined as follows:  $LS = l_1, \dots, l_k$ , where  $l_i \in L$  ( $1 \leq i \leq k$ ).
- IMC is the optimization function.

For a given L-PN LN and a label sequence LS of length p and m is the maximum of transition sharing the same label, the number of possible combinations to find a transition sequence to fire the label sequence is  $m^p$ , which is a polynomial complexity  $O(m^p)$ . We notice that the state space increases exponentially with p and m. Therefore to find the optimal value of MIM is an NP-hard problem [12].

The IMC value of a firing sequence FS is the computation of the number of tokens in each place p in the initial marking IM of FS:

$$IMC(FS) = \sum_{p \in P} IM(FS)[p] \quad (4)$$

### 3.2. ACO for MIM Computation Problem in L-PN

ACO is a population-based metaheuristic for solving difficult combinatorial optimization problems, where m solutions are generated at each iteration. Solutions are generated by a probabilistic constructive mechanism that is biased by numerical information called pheromones (and possibly by heuristic information) [23].

In our algorithm, we only consider the global communication by the pheromone information. The basic concepts of the proposed algorithm are:

- Types of individuals: admissible solutions obtained using a constructive algorithm.
- Type of evolution: generational replacement with a population of constant size.
- Information sources: search history (stored in the pheromone matrix).

#### 3.2.1. Solution Representation

The solution S for the MIM problem is a vector (sequence) of transitions of size n (n = length of LS), associated with an initial marking IM calculated with rule (3) and a number of tokens (objective function) IMC (4).

We denote  $S[i] = j$  the task number j will be executed in the  $i^{th}$  row in the S sequence. For example, for LS sequence of 8 labels="effabcd", a solution S can be the following sequence:

1	1	2	2	5	4	8	9
---	---	---	---	---	---	---	---

$S[3]=2$ , the transition T2 will be fired in  $3^{rd}$  place of firing sequence S.

The success of the ACO algorithm lies in its representation of the artificial pheromone. All the ACO exercise is devoted to maintain its pheromone artifice, the best solution is found based on trails of pheromone. In other words, the trail pheromone is a feasible solution, and the concentration of pheromone indicates the solution quality.

In our approach two types of pheromone matrices are used. The first is matrix  $\tau(m*n)$ , where m is a number of transition in the L-PN, n is the length of label sequence LS, and  $\tau_{ij}$  based on the experience of ants to fire the transition i in the  $j^{th}$  row

of the solution. The second matrix is  $\tau'(n*n)$ , where  $\tau'_{fk}$  concerns the experience of ant to cross the transition k after the transition f.

### 3.2.2. Solution Construction

Solution construction in ACO is made variable by starting from a random variable and the next variables are chosen in a probabilistic way. In our context, the ant's solution started with a random feasible transition according to the first label in LS.

Moreover, the decision for passing to one possible component transition ( $N_i$ ) by an ant k is made in formula (5), where each possible transition  $T_i$  has a probability  $P_{ij}$  of being chosen (transition i fired in  $j^{th}$  row in the solution). The transition which has the maximum probability will be chosen, where  $\tau_{ij}$  indicated the amount pheromone quantity in the arc (i, j) and  $\tau'_{ei}$  is the pheromone quantity in edge (e, i). The influence of the pheromone level  $\tau$  and is  $\tau'$  is controlled by the fixed parameters  $\alpha$  and  $\beta$  respectively which contribute together to the ant's solution construction.

The ACO rule to choose one element from  $N_i$  as follows:

$$P_{ij}^k = \frac{\tau_{ij}^\alpha \tau'_{ei}^\beta}{\sum_{i \in N_i^k} \tau_{ij}^\alpha \tau'_{ei}^\beta} \quad (5)$$

Where  $i \in N_i^k$ , N is the set of candidate transitions, j is the  $j^{th}$  row in the solution and e is the last transition passed by ant k.

---

#### Algorithm 1: Proposed ACO Solution Construction

---

**Input:** LN, LS

**Output:** Solution of ant k

```

1: AntSol=RandSeq (LS (1));
2: For i=2 to length (LS)
3:   e=AntSol (end);
4:   N=Next (LS (i));
5:   Prob= [];
6:   For v=1 to length (N)
7:     prob (v) =  $\tau(N(v), i)^\alpha * \tau'(e, N(v))^\beta$ ;
8:     transition (v) =N (v);
9:   End
10:  s= find (prob == max (prob));
11:  AntSol (i) = transition(s);
12: End

```

---



### 3.2.3. Pheromone Update

The indirect communication of artificial ants is modified by the pheromone matrix. The pheromone value represents the attractiveness of a specific edge for the ants. According to the experience gained at runtime, the higher the amount of pheromone on an edge is, the higher the probability that ants choose it when constructing solutions. Pheromone values are iteratively updated in two ways: pheromone evaporation and pheromone deposit.

The initial value of all arcs  $(i, j)$  of the pheromone matrix,  $\tau_{ij} = \tau_0$ , where  $\tau_0 > 0$ .

The formula of deposited pheromone in ACO:

$$\tau_{ij} \leftarrow \tau_{ij} + \sum_{k=1}^m \Delta\tau_{ij} \quad (6)$$

Where  $\Delta\tau_{ij}$  is the amount of pheromone deposited by ant  $k$  in the arc  $(i, j)$ , calculated with the following formula:

$$\Delta\tau_{ij} = \begin{cases} \frac{1}{IMC(S^k)}, & \forall (i, j) \in S^k \\ 0, & \text{else} \end{cases} \quad (7)$$

The algorithm would not be complete without the process of evaporating the pheromones' tracks. Indeed, to avoid being trapped in suboptimal solutions, it is necessary to allow the system to forget the bad solutions. The pheromone trail evaporation rule, with  $\rho$  an evaporation rate that can take value between 0 to 1:

$$\tau_{ij} \leftarrow (1 - \rho) \tau_{ij} \quad (8)$$

### 3.2.4. Rank-Based ACO

The particularity in rank-based ACO is on the deposit of pheromone, where only ant agents that have found one of some excellent solutions are allowed to regulate the pheromones. In the regulation of the pheromones' process, all ant agents are sorted by an ascending order of their fitness function ( $IMC_1 \leq IMC_2 \leq \dots \leq IMC_m$ ). At this time, the ants ranked between 1 and the  $\sigma$  ants become elite ants, where  $\sigma$  denote the number of ants having the same value of fitness function. Therefore, only the  $\sigma$  best ants contribute to the deposition of pheromones.

The concept of ranking is applied and extended to the ant system in our approach as follows: after all ants have generated a complete tour, the ants are sorted by their IMC and the contribution of an ant to the trail level update is weighted according to the rank of the ant.

---

#### Algorithm 2: Pseudo code of Rank-based ACO

---

1. Initialize a population of  $m$  ants.
2. Evaluate the  $m$  ants
3. While the stop condition is not satisfied, do
4. For  $k = 1$  to NumberOfAnts do

5. Build the path of the ant  $k$  (see Algorithm 1)
  6. End for
  7. Evaluate the  $m$  ants and rank them according to their fitness function
  8. Deposit pheromone on the trails of best ants
  9. Evaporate the pheromone trails
  10. End while
  11. Return the best solutions.
- 

#### 4. EMPIRICAL STUDIES

In this section, we improve the benefits of the proposed approach compared to the previous works.

##### 4.1. Empirical Settings

The proposed algorithm was developed in a MATLAB version 7.6 (R2016a) environment in a computer Intel(R) Core (TM) i5. 2.2 GHZ, 4 GB of RAM, SE windows 10 64 bytes and tested for different lengths of labels' sequences: 4, 8, 12, 16, 20, 24, 28, 32, 36 and 40 labels in the studied L-PN (Figure 1 in section 2). The ACO parameters are determined empirically.

The studied label sequence of length 40:

LS=effabddcbabbccaaddgghhdcbaeffabddcba.

We tested the performances of the proposed ACO algorithm by the tuned parameters values  $\alpha$ ,  $\beta$  (between 1 to 5), numbers of ants (10 to 100), the evaporation rate  $\rho$  (between 0 to 1). As the number of all possible transition sequences increases exponentially with the length of the label sequence, so the number of iterations has is proportional to the length of LS. For length LS (4 to 20) 200 sufficient iterations to achieve the optimal solution, while for length (24 to 40), there are at least 500 iterations to find the optimal solution.

The final fixed values in the proposed algorithm and their descriptions are recapitulated in Table 1.

**Table 1.** ACO parameters.

Parameter	Value
Weight on Pheromone Information $\tau$ ( $\alpha$ )	1
Weight on Pheromone Information $\tau'$ ( $\beta$ )	5
Pheromone Evaporation Rate ( $\rho$ )	0.5
Maximum N° of Iterations	500
Number of Ants	50
Initial pheromone value $\tau_0$	0.1

#### 4.2. Benefits of the Proposed Approach

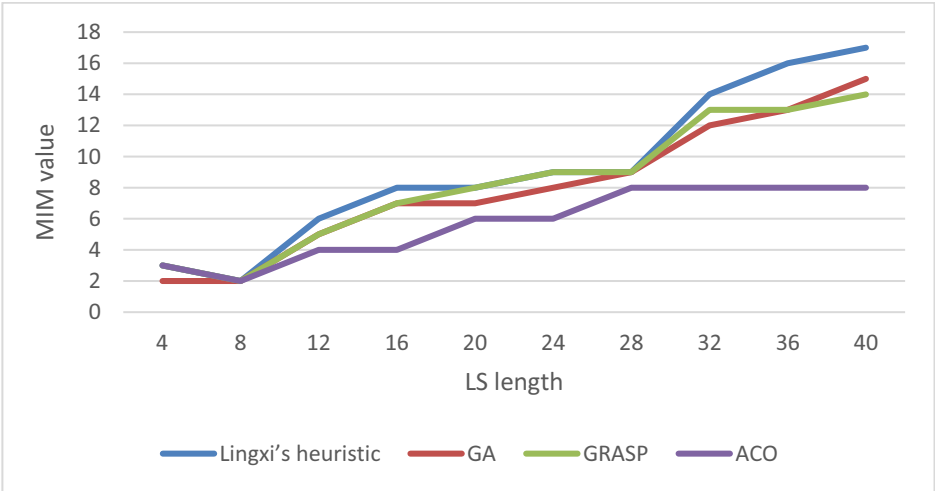
The performance of the proposed algorithm was tested on a set of 10 different LS lengths (4 ... 40) which have been tested in the previously proposed approach for the MIM computation problem.

An example for LS with length 40, a solution S, and the IM to fire the obtained S is presented in Table 2:

**Table 2.** Obtained solution.

<b>S</b>	[1 1 2 2 5 6 7 10 9 7 6 3 6 4 8 7 3 3 10 9 11 11 12 12 10 7 6 3 1 1 2 2 5 6 7 10 9 7 6 3]
<b>IM</b>	[3 0 2 0 0 1 2 0 0 0] (MIM=8)

The results are compared with those obtained by Lingxi’s heuristics [12], the genetic algorithm [14], and GRASP [13], and showed that our approach is better in terms of the solution quality. The results are recapitulated in figure 2.



**Figure 2.** MIM results of our proposed ACO based approach compared to previous works.

The run time of the proposed algorithms, depends on the ACO parameters, especially the number of iterations, and the number of ants. The benefits of our ACO-based approach in terms of run time is compared to others approaches as shown in Table 3:

**Table 3.** Computation time (s).

LS length	Lingxi's heuristic	GRASP	GA	Our ACO
Run Time (s)	21.235	3.7	4.692	5.922

#### 5. CONCLUSION AND FUTURE WORK

In this article, we focused on the MIM computation problem to minimize the resource consumption in a flexible production system modeled with Labelled PN, and we

proposed an approach based on ACO metaheuristic for solving this NP-Hard problem. The experimental results were running on several label sequence lengths and compared with published results available in the literature such as the genetic algorithm, the GRASP, and the heuristic function. Results show that the proposed algorithm gives good results in terms of MIM quality and computational algorithm run time.

There remains a lot of work to be done with the MIM computation problem in L-PN, especially in terms of execution runtime using other recent techniques of artificial intelligence.

## REFERENCES

- [1] Abdulziz. M., El-Tamimi, Mustafa H. Abidi, S. Hammad Mian b, Javed Aalam,” Analysis of performance measures of flexible manufacturing system “, Journal of King Saud University – Engineering Sciences 24, 115–129, 2012.
- [2] C. Vamsikrishna, G. Padmanabhan,” Role of Petri Nets in Flexible Manufacturing System – A Review”, International Journal of Engineering Trends and Technology (IJETT), 2016.
- [3] F Tüysüz, C Kahraman, “Modeling a flexible manufacturing cell using stochastic Petri nets with fuzzy parameters”, Expert Systems with Applications, 2010.
- [4] Chinnusamy.T.R, Karthikeyan.T, Krishnan.M, Rajaram.K, “Flexible Manufacturing System Scheduling Using Hybrid Petri Nets with Dispatching Rules”, International Journal of Mechanical & Mechatronics Engineering, IJMME-IJENS Vol:14 No:06, December 2014.
- [5] Bo Huang, R Jiang, G Zhang,” Search strategy for scheduling flexible manufacturing systems simultaneously using admissible heuristic functions and nonadmissible heuristic functions”, Computers & Industrial Engineering, 2014.
- [6] Bo Huang, ZhiCheng Cai, MengChu Zhou, JianGen Hao,” Scheduling of FMS based on Binary Decision Diagram and Petri Net”, IEEE 15th International Conference on Networking, Sensing and Control (ICNSC),2018.
- [7] B. Barzegar, H. Motameni, and H. Bozorgi, “Solving Flexible Job-Shop Scheduling Problem Using Gravitational Search Algorithm and Colored Petri Net,” Journal of Applied Mathematics, vol. 2012, pp. 1–20, 2012.
- [8] OT Baruwa, MA Piera,” Anytime heuristic search for scheduling flexible manufacturing systems: a timed colored Petri net approach”, The International Journal of Advanced Manufacturing Technology, 2014.
- [9] Zhao, Z., Zhang, G., & Bing, Z.,”Scheduling optimization for FMS based on Petri net modeling and GA”, 2011 IEEE International Conference on Automation, 2011.
- [10] J. P. Caballero-Villalobos, G. E. Mejía-Delgadillo, and R. G. García-Cáceres, “Scheduling of complex manufacturing systems with Petri nets and genetic algorithms: a case on plastic injection moulds,” The International Journal of Advanced Manufacturing Technology, vol. 69, pp. 2773–2786, Aug. 2013.
- [11] Lei, H., Xing, K., Han, L., Xiong, F., Ge, Z.,” Deadlock-free scheduling for flexible manufacturing systems using Petri nets and heuristic search “, Computers & Industrial Engineering (2014), doi: <http://dx.doi.org/10.1016/j.cie.2014.04.002>
- [12] L Li, CN Hadjicostis,” Minimum Initial Marking Estimation in Labeled Petri Nets”, IEEE Transactions on Automatic Control, 2012 - [ieeexplore.ieee.org](http://ieeexplore.ieee.org).
- [13] Hichem, K., Sliman, L., & Nabli, L. (2019, September).” GRASP-Based Approach for Minimum Initial Marking Estimation in Labeled Petri Nets”. In 2019 15th International Conference on Semantics, Knowledge and Grids (SKG) (pp. 75-80). IEEE.
- [14] H Kmimech, A. J Telmoudi, L Sliman , & L Nabli . (2020). “Genetic-based approach for minimum initial marking estimation in labeled Petri nets”. IEEE Access, 8, 22854-22861.
- [15] H Kmimech, AJ Telmoudi, L Sliman, L Nabli, “Minimum Initial Marking Estimation in Labeled Petri Nets using simulated annealing”,2019.
- [16] M Gam, A Abdellatif, AJ Telmoud, L Nabli, “Minimum Initial Marking Estimation in Labeled Petri Nets through Genetic Algorithm and Tabu Search”, 2020 4th International Conference on Advanced Systems and Emergent Technologies (IC\_ASET), 2020.
- [17] A Abdellatif, A J Telmoudi, Patrice Bonhomme & Lotfi Nabli, “Minimum Initial Marking Estimation of Labeled Petri Nets Based on GRASP Inspired Method (GMIM)”, Cybernetics and Systems, 51:4, 467-484,2020, DOI: 10.1080/01969722.2020.1770505
- [18] K. Ruan, L. Li, and W. Wu,”Minimum initial marking estimation in labeled Petri nets with unobservable transitions”, IEEE Access, vol. 7, pp. 1923219237, 2019.

- [19] G Calis\*, O Yuksel, "An Improved Ant Colony Optimization Algorithm for Construction Site Layout Problems", Journal of Building Construction and Planning Research, 2015.
- [20] M. Yousef\_khoshbakht, F. Didehvar F. Rahmati, "An Efficient Solution for the Vehicle Routing Problem by Using a Hybrid Elite Ant System", International Journal of Computers, Communications & Control (IJCCC), 2014.
- [21] S Tian, T Wang, L Zhang, X Wu," An Energy-Efficient Scheduling Approach for Flexible Job Shop Problem in an Internet of Manufacturing Things Environment", IEEE Access, 2019.
- [22] Kan Jun-man,Zhang Yi," Application of an Improved Ant Colony Optimization on Generalized Traveling Salesman Problem", International Conference on Future Electrical Power and Energy Systems,2012.
- [23] M López-Ibáñez, T Stützle, M Dorigo, "Ant Colony Optimization: A Component-Wise Overview". iridia.ulb.ac.be, 2015.