

An Improved Optimization Algorithm for Timed Petri Net Based on Genetic Algorithm

Li Tingpeng^a, Wang Nantian^b, Li Yue^c, Qian Yanling^d

Science and Technology on Integrated Logistics Support Laboratory, National University of Defense Technology, Changsha, 410073, China

^atovey1987@126.com, ^bwangnant@126.com, ^cliyue@nudt.edu.cn, ^dylqian@nudt.edu.cn

Keywords: Petri Net; Optimization; Genetic Algorithm; Production Scheduling

Abstract. A variety of theoretical and practical work exists on Petri-net-based optimization algorithm. However, the efficiency of optimization is still the bottleneck of its application, especial for NP problem (e.g., Flexible Manufacturing System (FMS) scheduling problem). In order to improve the optimization efficiency, an optimization approach for the Timed Petri Net (TPN) model is proposed in this paper. Firstly, definition of Simplified Timed Petri Net (Simplified TPN) and method of simplifying traditional model are proposed. Based on this Simplified TPN, the concepts of mutual exclusion contracts and order contracts are defined, and the optimization mathematical model is obtained. Then, a reachability checking method is introduced to ensure the feasibility of solutions. Finally, this paper proposes a new crossover operator and mutation operator of Genetic Algorithm (GA) and describes the steps of optimization approach. This proposed approach is evaluated on FMS scheduling.

1 Introduction

Petri net is a graphical and mathematical modeling and analyzing tool for systems characterized as concurrent, distributed, asynchronous, parallel and/or stochastic[1]. The concept of Petri net was developed and introduced by Carl Adam Petri in 1962[2].

In the literature, there have been some approaches to be presented to study the optimization problem based on TPN with GA. For example, Hao Dong et al. proposed an approach to optimize FMS Scheduling Problem based on improved Genetic Algorithm (GA) and Petri nets model[3]. Zhang X. X et al. proposed an encoding /decoding mechanism to simplify the Genetic Algorithm and to optimize the JSP and FMS scheduling problem[4]. In order to solve the state space explosion problem, Andre Bevilaqua et al. defined a group of reduction rules and created a GA to automatically reduce TPN models[5]. In[6], A multi-objective scheduling method based on the controlled Petri net and GA was proposed for the dynamic JSP constrained by machines, workers. Later on, an improved GA is proposed and applied in the Job-Shop Scheduling Problem[7] to overcome the weakness of premature convergence appearing in GA. Chen W.M et al. presented an extended timed Petri Net (ETPN) based Parthenon GA to solve the Job-Shop Scheduling problem[8]. In[9], a GA embedded search strategy over Hierarchical Colored Timed Petri Net (HCTPN) for semiconductor wafer fabrication was proposed. Albert et al. proposed a Petri net and GA scheduling method for job shop manufacturing systems, in which Taguchi's method was used to optimize the parameters of GA to improve the optimization results[10]. Bikram Sharda et al. presented an approach using multi objective GA, Petri nets and Bayesian model averaging (BMA) for robust design of manufacturing systems. The objective function aims at minimizing make span, mean WIP and number of machines, while considering uncertainties in processing times, equipment failure and repairs, and product demand[10]. A novel knowledge based GA for path planning of multiple robots for multiple targets seeking behavior in presence of obstacles is proposed in[11]. O. Morandin Jr. et al. improved GA and Timed Petri Net to optimize the FMS scheduling problem. This approach considers the input buffers of machines, AGVs, and flags to avoid deadlock[12].

Although the optimization theory and application of Timed Petri Net have been developed greatly, the optimization efficiency for NP problem of which computational cost grows

exponentially with the size of problem space[13], such as FMS Scheduling Problem and JSP, still needs to be improved. In this paper, an approach is proposed to improve the optimization efficiency of GA based on TPN.

2 The optimization mathematical model based on Simplified TPN

2.1 Simplified Timed Petri Net model.

Model is the base of an optimization algorithm which affects the efficiency of optimizing significantly. In order to reduce the complexity of TPN model, Simplified Timed Petri Net (Simplified TPN) is proposed.

Definition 1: Simplified TPN

The Simplified TPN model refers to the TPN model which satisfies the following two conditions.

Condition 1: Every operation process is modeled by one transition.

Within traditional TPN, every operation process is modeled by two transitions and one places. However, in the model of simplified TPN, an operation is modeled by one transition and using double sided arcs to link resource places and the operation transition. Fig. 1, Fig. 2 and Fig. 3 illustrate the convert from traditional TPN of classical operation process in FMS.

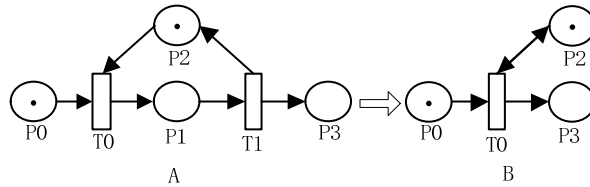


Fig. 1 The operation process with single machine

The model of Fig. 1 A represents the traditional TPN model of the operation process with a single machine. The Simplified TPN is shown in Fig. 1B and the simplification procedure is depicted as following:

Step1, delete the process place *P1* and the ending transition *T1*

Step2, use double sided arcs to link resource place *P2* and *T0*.

Step3, add the time parameter to *T0* from *P1*.

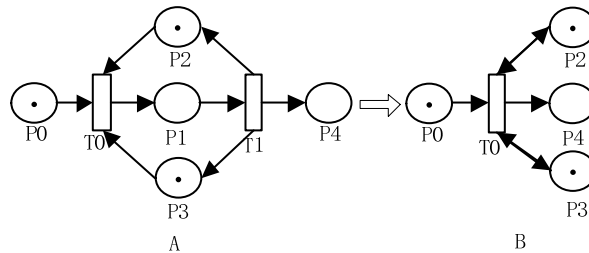


Fig. 2 The operation process with multi machines

Fig. 2A models the operation process with multi machines (2 machines in the example (*P2* and *P3*)), which means the operation needs multi machines when running. The simplified model is shown as Fig. 2B.

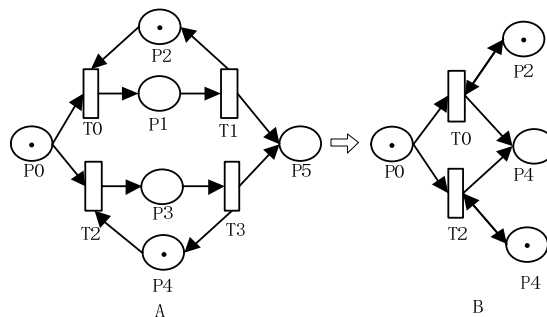


Fig. 3 The operation process with selectable machines

Fig. 3A models the operation process with selectable machines, which means the operation can

start with any of the multi machines. The Simplified TPN is shown in Fig. 3B.

Condition 2: The frequency of transitions being fired is not more than once.

The essence of many optimization problems based on Petri nets, such as FMS scheduling problem, routing planning and JSP, is to find the best path from the initial marking to the target marking and the path consists of sequence of transitions. In order to reduce the complexity of coding of the sequence of transitions, the traditional TPN model is simplified by separating transitions and adding initial places to limit the firing times of each transition. An example is shown as Fig. 4.

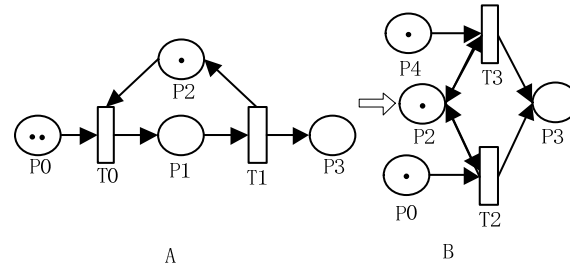


Fig. 4 The Simplified TPN of operation process with two work pieces

In Fig. 4A, the two tokens translate from $P0$ to $P3$, transition $T0$ and $T1$ must be fired twice. While, Fig. 4B models the same process, but each of transitions is fired only once.

The simplification procedure of the traditional TPN model is shown as following:

Step1, use two sided arcs and operation transition to replace the traditional operation model (transition pair and operation place) with the steps introduced in **Condition 1**.

Step2, if transitions may need to be fired more than once, simplify the transitions with the method introduced in **Condition 2**.

An example of Simplified TPN is shown in Fig. 5. The model represents a FMS scheduling system, which includes two work pieces (J1 and J2) and six machines. In the model, P1 and P6 represent that work piece J1 and J2 are ready to be machined, respectively. M1, M2, M3, M4, M5 and M6 represent six individual machines, respectively. J1 needs four operation processes before being finished, and J2 needs three operations at least.

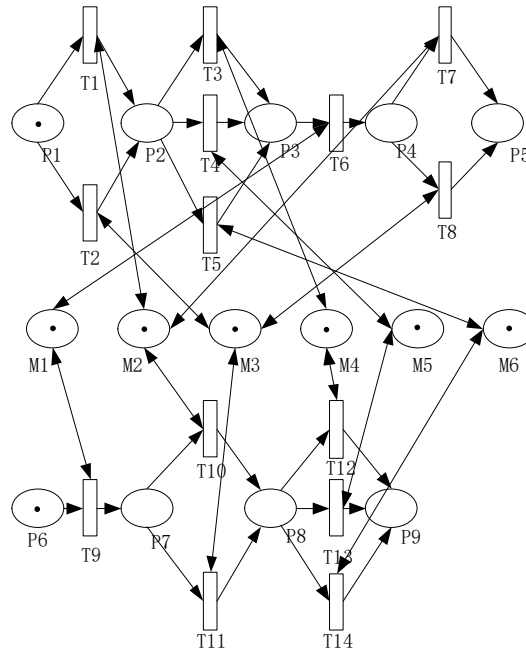


Fig. 5 An example of simplified TPN

2.2 Optimization mathematical model

Generally, the solution of the simplified TPN is a sequence of transitions. In order to define the fired order of transitions in the sequence, two principles are proposed:

Principle1, the former transitions are in the sequence, the earlier are fired.

Principle2, All the closed transitions should be fired at the same time if they are enabled.

For example, **Seq 1** ($T_2, T_5, T_3, T_1, T_4, T_6, T_8, T_{14}, T_{12}, T_7, T_9, T_{11}, T_{13}, T_{10}$) is one of the transition sequences of the Simplified TPN (shown in Fig. 5). According to principle1, T_2 is fired earlier than T_5 , T_5 is fired earlier than T_3 , ..., T_{13} is fired earlier than T_{10} . However, T_7 and T_9 must be fired at the same time according to principle2, because T_7 and T_9 are closed in the sequence and are fired simultaneously.

Theoretically, there are $A_{N_T}^{N_T}$ solutions for a TPN model, where N_T is the number of transitions. However, not every sequence of transitions is a reachable solution, such as **Seq 1**, because T_{13} cannot be fired earlier than T_{10} . A reachable sequence has to satisfy some conditions. In this paper, mutual exclusion contracts and order contract conditions are proposed to check the reachability of sequence.

2.2.1 Contract conditions

In order to easily check the reachability of sequence and reduce the complexity of coding space, two contracts (mutual exclusion contracts and order contracts) are defined. Mutual exclusion contracts check whether two transitions are in a same sequence or not, and order contracts check the firing order of transitions.

2.3 Mutual exclusion contracts

Definition 3: Mutual exclusion contract C_m refers to those transitions which have the same input and output places without considering resources places.

For Simplified TPN of FMS scheduling problem, only one transition in a mutual exclusion contract is fired in a certain scheduling plan at the same time. For example, in Fig. 6, T_0 and T_1 constitute a mutual exclusion contract. They represent a same operation process with different equipment, but only one of them can be fired at one time. We use (T_0, T_1) to denote the mutual exclusion contract.

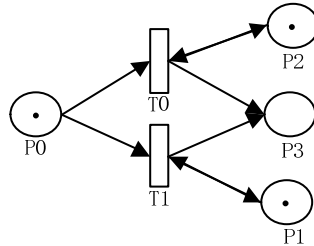


Fig. 6 An example of mutual exclusion contract

M_m represents all the mutual exclusion contract of the simplified TPN:

$$M_m = \{T_{N1}, T_{N2}, \dots, T_{Nk}\}$$

Where,

$$T_{Ni} = (T_{i1}, T_{i2}, \dots, T_{ip}) \quad i=1, 2, \dots, k$$

T_{Ni} represents certain mutual exclusion contract, p is the number of transitions of T_{Ni} and k is the number of mutual exclusion contracts of the simplified TPN.

The steps of determining the mutual exclusion contracts of simplified TPN are shown as following:

Step 1, Remove the resource places from the Simplified TPN which link to transitions with two sided arcs, such as place P2 in Fig. 7 A.

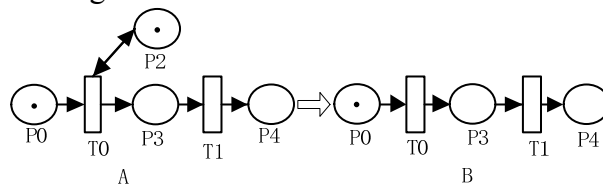


Fig. 7 Remove the resource place from Simplified TPN

Step 2, Find all the input places and output places of each transition.

Step 3, Find out the transitions which have the same inputs and outputs place. Those transitions make up of a mutual exclusion contract.

2.3.1.1 Order contracts

Definition 2: Order contract C_o , consisting of two transitions, defines the fired order of transitions. For example $C_{oi} = (T_1, T_2)$ means T_1 must be fired earlier than T_2 , i.e., T_1 must be ahead of T_2 in the sequence of transitions. For a Simplified TPN, generally, there are more than one order contract in the model. M_o represents the order contracts of certain Simplified TPN:

$$M_o = \{ C_{o1}^T, C_{o2}^T, \dots, C_{on}^T \}$$

where C_{oi}^T ($i=1,2,\dots,n$) is the transposition of C_{oi} and n is the number of contracts.

In order to calculate the order contracts automatically, we propose the following method to obtain order contracts of Simplified TPN:

Step 1, Remove the resource places of simplified TPN with the method introduced in Fig. 7.

Step 2, Find all the input places and output places of each transition. If one of the output places in transition A is one of the input places of transition B, then A should be fired earlier than B and (A, B) is an order contract, such as (T0, T1) in Fig. 7 B.

Step 3, Search all the transitions and find out all the order contract of the Simplified TPN.

2.3.2 Target function

For many FMS scheduling problems, reducing the scheduling time is the target of optimization. Therefore, the total time (T_{total}) of the scheduling plan (one chromosome in GA, i.e., a sequence of transitions) is regarded as a target function.

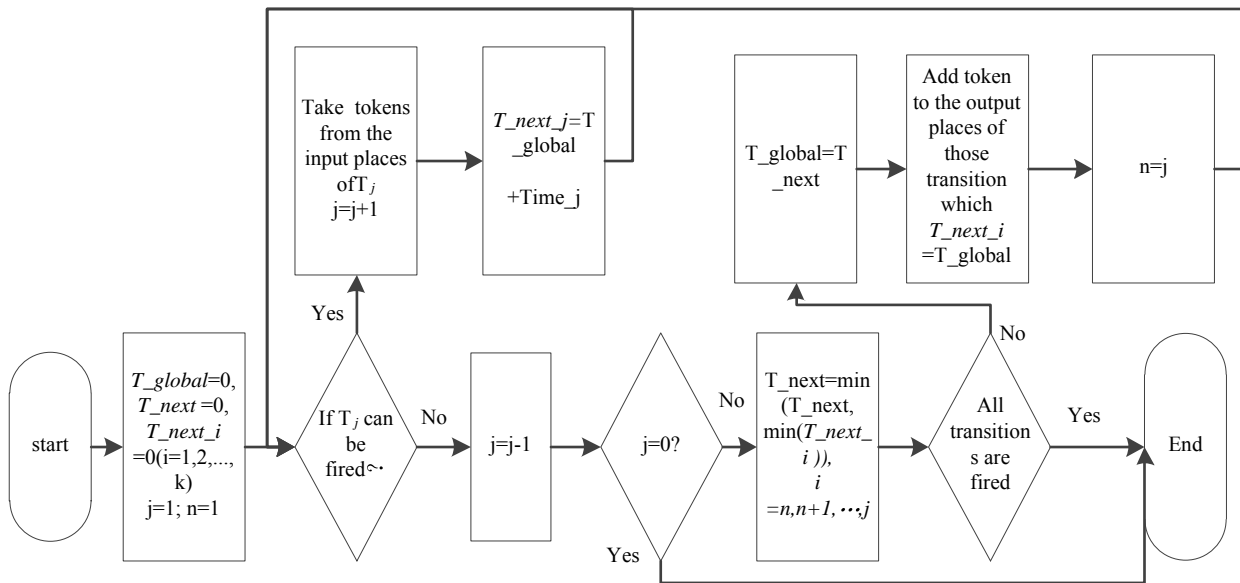


Fig. 8 The flow graph of calculating T_{total}

Based on the simplified TPN, it depicts a way to calculate T_{total} as shown in Fig. 8 :

Step 1, the global time is Initialized as zero $T_{global}=0$, the next global check time $T_{next}=0$ and the next check time of each transition in the sequence $T_{next_i}=0$ ($i=1, 2, \dots, k$). Set, $j=1$ and $n=1$.

Step 2, if transition j (T_j) cannot be fired, go to **Step 3**. Otherwise, the tokens of the input places of T_j will be removed and next check time T_j will be updated as:

$$T_{next_j} = T_{global} + Time_j$$

Then, $j=j+1$ and continue **Step 2**.

Step 3, $j=j-1$, if j equal to zero, fail to calculate and go to **Step 5**, otherwise:

$$T_{next} = \min(T_{next}, \min(T_{next_i}), i = n, n+1, \dots, j)$$

Step 4, if all transitions are fired, goes to **Step 5**, other wise

$$T_{global} = T_{next}$$

Add tokens to the output places of those transitions which T_{next_i} equals to T_{global} , Then $N=j$ and go to **Step 2**.

Step 5, Stop calculating. If the calculation is successful, the T_{total} can be acquired:

$$T_{total} = T_{global}$$

2.3.3 Mathematical model

Based on the coding mechanism, contracts conditions and target function, the mathematical model of the simplified TPN is shown as following:

$$\begin{cases} \min T_{total}(T_n) \\ M_0 \\ M_m \end{cases}$$

where, $T_{total}(T_n)$ is the total time of the transition sequence T_n , M_o denotes order contracts conditions and M_m denotes mutual exclusion contracts conditions, and T_n is the transition sequence (an scheduling plan).

3 The improved GA based on Simplified TPN

Genetic algorithms (GA) are heuristic stochastic search methods based on biological evolution mechanisms^[11]. GA uses several genetic operators and strategies to find better solutions for a certain problem by selectively exploring new regions of the solution space with a set of individuals. Crossover and mutation are the most common genetic operators, which emulate the reproduction and spontaneous alteration of individuals in living species^[16]. An individual, also known as a chromosome, represents a solution to the problem and is usually coded by binary^[17] or integer^[3]. The set of individuals used by GA is called a population. Implicit parallelism and global searching ability are the main characteristics of GA. Therefore, GA is suitable for optimizing complex NP, such as FMS scheduling problem. In this paper, an improved genetic algorithm based on the Simplified TPN is proposed via improving the genetic operators and simplifying the reachability checking with the contract conditions of the mathematical model.

3.1 Chromosome coding mechanism

Coding is the first step of Genetic Algorithm and has significant impact on evolution efficiency. According to the mathematical model of simplified TPN, the sequences of the transitions are used as chromosomes, and the integer coding mechanism is used in this paper.

Taking Simplified TPN shown in Fig. 5 as an example, the chromosome of **Seq 1** is **Chro1**:

Chro1 (2, 5, 3, 1, 4, 6, 8, 14, 12, 7, 9, 11, 13, 10)

The locations of the number in this chromosome represent the fired orders of transitions and all the genetics (an integer in chromosome) obey the two principles described in 2.2.

3.2 Reachability checking

Typically, not all the sequences of transitions are reachable chromosomes (schedule plans, for FMS scheduling problem). In order to determine the reachability of a chromosome, the general approach is to calculate the fitness (scheduling time, for FMS scheduling problem) or simulate it and determine whether it can finish the task. Since there are many unreachable chromosomes of NP problems, this approach costs a large number of calculation, which hampers the efficiency of evolution. In order to reduce the computation quality of GA, based on the contract conditions of Simplified TPN, we propose a new approach to check and to modify the reachability of chromosome. Furthermore, a method is proposed to modify the chromosome to be reachable.

All the reachable chromosomes of Simplified TPN must satisfy the mutual exclusion contracts and order contracts.

3.2.1 The Steps of checking mutual exclusion contracts

According to the definition of mutual exclusion contracts, only one of transitions in a contract

can be fired in a chromosome. In order to satisfy the mutual exclusion contract and not destroy the multiformity of chromosome, the first of the transitions in the chromosome of each mutual exclusion contract is remained and the rest transitions will be deleted. The method is shown as following in detail.

Step 1, Find out the mutual exclusion contracts with the steps shown in 2.3.

For example, the mutual exclusion contracts of the Fig. 5 model are shown in Table 1.

Table 1 The mutual exclusion contracts of Fig. 5 model

No.	contract	No.	contract	No.	contract
1	(1,2)	2	(3,4,5)	3	(7,8)
4	(10,11)	5	(12,13,14)		

Step 2, Take one set of mutual exclusion contract into consideration. For example, take No.1 (1, 2) into consideration.

Step 3, Search and determine the location of each transition of the mutual exclusion contract in the chromosome and delete all the transitions of the mutual exclusion contract except for the first transition in the chromosome.

The checked chromosome of **Chro1** is **Chro1***(only No.1 is checked):

Chro1 (2, 5, 3, 1, 4, 6, 8, 14, 12, 7, 9, 11, 13, 10)

Chro1* (2, 5, 3, 4, 6, 8, 14, 12, 7, 9, 11, 13, 10)

Step 4, Go to **step 2** until all the mutual exclusion contacts are iterated.

All the mutual exclusion contracts are checked and the checked chromosome is **Chro1****:

Chro1** (2, 5, 6, 8, 14, 9, 11)

3.2.2 The steps of checking the order contracts

The order contracts define the fired order of transitions and the reachable chrome must satisfy the contracts. The procedure of checking and modifying the order contracts are described as following

Step 1, Find out the order contract matrix with the steps shown in 2.3.

For example, the order contracts of Fig. 5 model are shown as Table 2.

Table 2 The order contracts of Fig. 5 model

No.	contract	No.	contract	No.	contract	No.	contract
1	(1,3)	2	(1,4)	3	(1,5)	4	(2,3)
5	(2,4)	6	(2,5)	7	(3,6)	8	(4,6)
9	(5,6)	10	(6,7)	11	(6,8)	12	(9,10)
13	(9,11)	14	(10,12)	15	(10,13)	16	(10,14)
17	(11,12)	18	(11,13)	19	(11,14)		

Step 2, Take one of the order contracts into consideration, e.g. No.19 (11, 14).

Step 3, If the locations of the transitions in the chromosome do not satisfy the order contract, exchange the location of the transitions. **Chro1***** is the checked chromosome of **Chro1**** (only consider No.19):

Chro1** (2, 5, 6, 8, 14, 9, 11)

Chro1*** (2, 5, 6, 8, 11, 9, 14)

Step 4, Go to step 2 until all the order contracts are iterated.

All the order contacts are checked and the checked chromosome is **Chro1******:

Chro1**** (2, 5, 6, 8, 9, 11, 14)

Verifiably, the chromosome of Simplified TPN which satisfied the two contracts must be reachable, such as **Chro1******.

3.3 Evaluation operator

3.3.1 Crossover operator

The general crossover operator refers to the method to generate the child chromosomes from the two parent chromosomes by exchanging some genetics randomly according to certain crossover probability. However, for the Simplified TPN, the general crossover operator will destroy the

reachability of chromosome and cannot be modified only by reachability checking. Therefore, a new crossover operator is proposed.

The main function of crossover is to keep the good genetic of parent chromosomes and extend the multiform of chromosomes. Considering the characteristic of the simplified TPN, an improved crossover operator, which exchanges the genetic within one parent chromosome, is proposed. The crossover operator is described in detail as following:

Step 1, Generate some exchange location pairs randomly and the number of location pairs are less than half of the number of the genetic of parent chromosome, such as (1, 4) (3, 6,) in **Chro1******.

Step 2, Exchange the genetic of the parent chromosome.

Chro1**** (2, 5, 6, 8, 9, 11, 14)

Chro1***** (8, 5, 11, 2, 9, 6, 14)

where **Chro1******* is the child chromosome generated by Crossover operator.

Step 3, Check the reachability of the child chromosome.

Chro1***** (8, 5, 11, 2, 9, 6, 14)

Chro1***** (5, 2, 9, 6, 11, 8, 14)

where **Chro1******* is the reachable child chromosome of **Chro1******.

3.3.2 Selection operator

The selection operator is to keep the good chromosomes and to generate next population. Roulette and sort are general selection operators. In this paper, sort method is chosen to preserve those chromosomes which own good fitness. The sort method is to sort the chromosomes according to the fitness value in descending order and to choose individuals with high fitness.

In order to avoid the premature convergence problem caused by sort selection method, the partial replacement strategy with generation gap is merged into the selection operator. Therefore, the child generation can be obtained with following method: Firstly, the keep scale is P_r and the number of the parent generation is NOF . The $NOF * P_r$ of the sorted parent chromosomes which have bigger fitness are chosen into child generation. Secondly, the rest $NOF * (1 - P_r)$ chromosomes are generated randomly and are putted into the child generation after being checked the reachability.

Mutation operator. The mutation operator which changes some genetic of a chromosome randomly is introduced to make the GA has the ability of random searching in local field^[21, 23]. In order to keep the reachability of chromosomes, a new mutation operator is presented in this paper. This mutation selects the mutual exclusion contracts transitions as mutation point and replaces it with other transition in the mutual exclusion contracts randomly. The method is described as following:

Step 1, Select the number and location of mutation point of a chromosome randomly. For example, the number is 3 and the location is (1, 4, 5).

Step 2, Replace the genetic of mutation point with the next transition in the corresponding mutual exclusion contracts.

Chro1**** (2, 5, 6, 8, 9, 11, 14)

Chro1***** (1, 5, 6, 8, 9, 10, 15)

The mutated chromosome satisfies the order and mutual exclusion contracts. Therefore, it still preserves the reachability.

Fitness function. Fitness is the evaluation criterion of the quality of chromosomes, which will affect the optimization efficiency. Simple and exact calculation approach is very important for fitness. In order to get the minimum scheduling time, global time is introduced to calculating fitness and making as many transitions as possible fired paralleling is the most important principle.

$$f_i = 1/T_{total}$$

where f_i is fitness of chromosome i , T_{total} is the minimum scheduling time of the chromosome and can be calculated with the steps illustrated in 2.3.2.

Optimization approach. Based on the simplified TPN and the improved Genetic Algorithm, an

optimization approach is proposed in this paper as shown in Fig. 9.

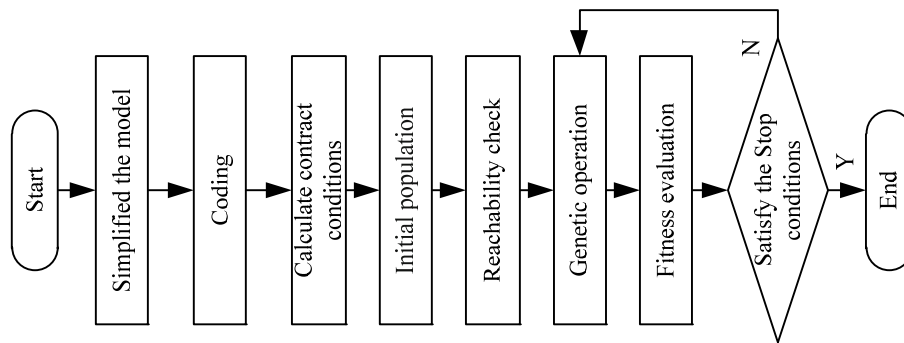


Fig. 9 The flow chart of the optimization approach

Step 1, Simplify the model with the approach introduced in section 1.2.

Step 2, Code each transition of the Simplified TPN with integer coding mechanism introduced in 3.1.

Step 3, Calculate the order contracts and mutual exclusion contracts of the simplified TPN with the steps of 2.2.1. Based on the contracts, the mathematical model can be obtained with the target function defined in 2.3.2.

Step 4, Initialize the generation of chromosomes randomly.

Step 5, Check the reachability of the initial population. As introduced in 3.2, the reachability checked chromosome has fewer transitions and represents a reachable schedule plan.

Step 6, Use the evaluation operator to generate the next generation chromosomes.

Step 7, Calculate the fitness of each chromosome.

Step 8, If the optimization approach satisfies the stop conditions, then stop, otherwise go to **step 6**.

4 Summary

In order to improve the optimization ability of Petri Nets, a mathematical optimization model containing mutual exclusion contracts and order contracts is established in this paper to improve the Genetic algorithm. Firstly, the simplified TPN is defined and the method of simplifying traditional TPN is described. Secondly, according to the simplified TPN, mutual exclusion contract and order contracts are defined and the calculation method is presented. Based on the contracts, the mathematical optimization model is acquired. Thirdly, in order to improve the search speed and to reduce the calculation cost, an improve Genetic Algorithm is proposed to optimize the simplified TPN. This proposed optimization approach is applied in a JSP to evaluate its feasibility. Results show that the proposed method is not only able to find better solutions but also more efficient.

References

- [1] Tadao Murata. Petri Nets: Properties, Analysis and Applications[J]. PROCEEDINGS OF THE IEEE, 1989, 77(4): 541-580.
- [2] Bijan Sarkar, Bimal Samanta. Application of Petri nets for systems modelling and analysis [J]. OPSEARCH, 2012, 49(4): 334-347.
- [3] HAO Dong, JIANG Chang-Jun, LIN Lin. Petri Net Based Modeling and GA Based Scheduling for FMS [J]. CHINESE JOURNAL OF COMPUTERS, 2005, 28(2):201-208.
- [4] ZHANG Xin-xu, ZHANG Wan-liang. Timed Petri Net Based Modeling and GA Based Scheduling of Flexible Manufacturing System [J]. Systems Engineering, 2010, 28(11):86-94
- [5] Edilson R. R. Kato Andre Bevilaqua, Emerson C. Pedrino, Roberto H. Tsunaki. Automated Generalized Petri Net Reduction Using an Evolutionary Approach Applied to a Manufacturing System Model[C]. 2013 IEEE International Conference on Systems, Man, and Cybernetics,

2013:2360-2365.

- [6] TAO Ze, LI Xiao-jun, LIU Xiao-xia. Petri Net and GA Based Approach for Dynamic Multi-objective JSP [J]. Modular Machine Tool&Automatic Manufacturing Technique , 2011, 2011(10):05-09.
- [7] Tang Tao. Research On JSP Scheduling Method Based On Petri Nets and Genetic Algorithm [D]. SOUTHWEST JIAOTONG UNIVERSITY , 2009.
- [8] CHEN Wei-min, WANG Bo, WEI Yu-ke. Research on Petri Net Based Genetic Algorithm in Job—Shop Scheduling Problem [J]. JOURNAL HARBIN UNIV.SCI.&TECH, 2008, 13(1):59-62.
- [9] CAO Zheng-cai, YU Hong-xia, QIAO Fei. Petr-Net and GA-Based Approach to Modeling and Optimize for Semiconductor Wafer Fabrication [J]. ACTA ELECTRONICA SINICA , 2010, 38(2):340-343.
- [10] Amarnath Banerjee Bikram Sharda. Robust manufacturing system design using multi objective genetic algorithms,Petri nets and Bayesian uncertainty representation[J]. Journal of Manufacturing Systems, 2013, 2013(32):315-324.
- [11] Dayal Ramakrushna Parhi Jagadish Chandra Mohanta, Saroj Kumar Patel. Path planning strategy for autonomous mobile robot navigation using Petri-GA optimisation[J]. Computers and Electrical Engineering, 2011, 2011(37):1058-1070.
- [12] E.R.R. Kato O. Morandin Jr., C.C.M.Tuma. A strategy of production scheduling with the fitness function of genetic algorithm using Timed Petri net and considering AGV and the input buffer[J], 2010:1311-1316
- [13] S.H. Chung F.T.S. Chan, P.L.Y. Chan, G. Finke, M.K. Tiwari. Solving Distributed FMS Scheduling Problems Subject to Maintenance: Genetic Algorithms Approach[J]. Robotics and Computer-Integrated Manufacturing, 2006, 2006(22):493-504.
- [14] YANG Zheng-lei, SONG Jian-she, CAO Ji-ping, YE Qing. D1SPN model of missile army maneuvering combat process [J]. Systems Engineering and Electronics , 2012, 34(5):936-940.
- [15] M. Ghaeli S.A. Sadrieh , P.A. Bahri, P.L. Lee. An integrated Petri net and GA based approach for scheduling of hybrid plants[J]. Computers in Industry, 2007, 2007(58):519-530.
- [16] Juan Pablo Caballero-Villalobos, Gonzalo Enrique Mejía-Delgadillo, Rafael Guillermo García-Cáceres. Scheduling of complex manufacturing systems with Petri nets and genetic algorithms: a case on plastic injection moulds[J]. Int J Adv Manuf Technol, 2013, 2013(69):2773-2786.
- [17] JIANG An, WANG Jin-shan. Gray Combination Forecast Of Equipment Support Cost Based On Genetic Algorithm[J]. Command Control & Simulation , 2011, 33(3):65-67.