Available online at www.sciencedirect.com

SciVerse ScienceDirect

Energy Procedia 17 (2012) 506 – 512

Energy Procedia

2012 International Conference on Future Electrical Power and Energy Systems

Hybrid Heuristic Search Based on Petri Net for FMS Scheduling

Qiushuang Wang[a] ,Zhongdong Wang[b]

Department of agriculture JiLin University 5333, XiAn Road, ChangChun City JiLin Province,130062,P.R.China,

[a]25501120@qq.com, [b]wangzd11@sohu.com

Abstract

This paper describes FMS scheduling problem based on Petri net with the goal of minimizing makespan . A hybrid heuristic search approach for the scheduling is presented, which combine dynamic search window with best-first algorithm and backtracking search., In addition redundant markings can be eliminated to reduce the searched state space. We use the proposed method to search for an optimal schedule of an example with varied lot sizes for each job type considered. Comparison with previous work is given to show the proposed approach can generate an optimal solution, but with much less search time and memory space. This hybrid algorithm can be deal with large systems.

Keywords : Flexible manufacturing system；scheduling；Petri net；heuristic search.

I.Introduction

Flexible manufacturing system (FMS) is an automatic production environment, and there may be several parallel production environments in it. Different products can be processed simultaneously, and many resources can be shared. FMS has many important properties, such as concurrent, shared, routing flexibility, batch variable, etc. It is difficult to model the complexity such a system showed by using traditional technology. Scheduling is the most important and difficult in operation of FMS, which belongs to NP-Hard problem [1, 3, and 4]. For this problem, many methods have been proposed, such as queuing theory, mathematical programming, system simulation, control theory, artificial intelligence, etc. [2].

Petri net, as a formal tool of the combination of mathematical analysis and graphics features, is used to describe the characteristics of concurrent, resource sharing, machining path diversity, etc. And also the deadlock can be effectively solved, which is considered as the effective tools of modeling, analysis and

 doi:10.1016/j.egypro.2012.02.128

monitoring for FMS [3, 4, 8, 12]. Petri net has also been used in the study of FMS scheduling problem recent years, Shih and Sekiguchi [4] used local scheduling to solve conflict during the using of Petri net simulation FMS; based on the scheduling performance evaluation obtained by the Petri net simulation running, Wang Xiaorong, et, al. [5] used two-stage hierarchical evolutionary optimization approach to solve the optimization scheduling problem of flexible production; Zhang [6] used Petri net to model the production assembly process, and combined dynamic programming approach to solve scheduling problem; Xie[7] used the combination of genetic algorithm (GA) Petri net to solve the configuration variable production line scheduling problem, Pan Quanke [8] used the hybrid algorithm of the combination of genetic algorithms and simulated annealing hybrid algorithm to optimize the scheduling of green manufacturing model. According to the combination of heuristic algorithm and Petri net model, it produced and searched reachability graph of part state, and then to obtain the optimal or sub-optimal scheduling scheme, which was the main method to solve the Petri net-based problem [3]; Lee and DiCesare[3] used A * algorithm for the Petri net-based production scheduling problem; Sun and Chen [9] improved Lee's method, and proposed the A * algorithm of limited expansion; Xiong and Zhou[10] proposed hybrid algorithm which combines stage backtracking and Best-First; Jeng[11] used approximation of state equation to establish efficient heuristic function; Moro, et al.[12] used dynamic window search technology in scheduling optimization；Gu Tianlong[13] proposed the representation of ordered binary decision diagram with related data in state space and the search process, avoiding the display enumeration of the state and search. All of the algorithms may be used to improve the efficiency by increasing the missing of optimal solution probability, or the generality of its approach is not enough. When the system scale is increased, the vast majority of methods often finally become a kind of similar breadth search algorithm with low efficiency [14].

Based on timed Petri net model of FMS, and taking the make span as its goal, the hybrid heuristic algorithm that combines dynamic window technology with Best-First algorithm and backtracking method is proposed in this paper. Simultaneously, the characteristics of use changing concurrent should be used to delete the redundant identity, and further to narrow search space. This algorithm is applied to the instance, and compares with the other results, verifying the effectiveness of this algorithm.

2.The Establishment of Model

2.1.Timed Petri net

Timed pool Petri net (TPPN) can be expressed as the following 6 tuples.

$\mathrm{TPPN} = (P, T, F, W, M_0, DT)$ ， in which

$P = \{p_1, p_2, \ldots, p_m\}$ is a finite pool set, expressing resources or operations;

$T = \{t_1, t_2, \ldots, t_n\}$ is finite changing set, expressing the event;

$F \subseteq (P \times T) \cup (T \times P)$ is directed arc set;

$W : F \to \{1, 2, 3, \ldots\}$ is a mapping function of weight;

$M_0 : P \to \{0, 1, 2, \ldots\}$ is initial marking, expressing the initial state of system

$DT : P \to R^+$ is the time set that related to pool. It need a certain time of each change from enabling to triggering, namely, the maximum operating time it enters to the pool.

2.2.Model description

Model assumption: multiple products can be processed and handled in the system. Each product has its own processing order, and it can also have a variety of orders. There is no order between different products. Multiple resources may be required in each product, but resources are exclusive and non-preemptive in the manufacturing process.

Search space: the state space of system Petri net model;

Search goals: find the change activation order that makes the smallest evaluation function f (M) change from initial marking m0 to target marking mF.

In which: f(m)= g(m)+ h(m);

g (m) is the determined makespan of partial scheduling from m0 till now, h (m) is the estimated value of makespan from now to target marking mF. Namely the next marking should be selected and determined through f (m).

3.Scheduling Algorithm

3.1.Search window

In search window SF, top-level and bottom-level is the maximum and minimum depth of the marking it contains. Search window only can accommodate depth (m) between [top-bottom] (depth (m) is the depth of marking m in reachability tree), after the searching of marking in the bottom, the search window must be pushed on. The marking number that the layer of l depth contains is size (l-depth), and it can't more than max-node. Generally, it uses [top-level, max-node, window-size] as parameters to definite search window. There is a limit set by search window for the search space of heuristic function, avoiding the index increasing of the state space.

3.2.Scheduling algorithm

1) Put initial marking m0 into the open table;

2) If open table is empty, then to return error, and end algorithm;

3) Obtain a $f(m)$ minimum marking m from open table:

If it is target state, then return to the constructed optimal path from initial state to the target state, and end algorithm;

If top-depth <depth (m), then obtain a depth (m) = bottom-depth and minimum marking m of f (m) from open table, and move it to close table; if size (bottom- depth) = max-node, then the search window will move forward;

If the marking of m is bottom-depth≤depth(m)<top-depth，size(depth(m))<max-nod, then move it to close table;

If bottom-depth>depth (m), then move this marking from open table;

4) Find out all the changes can be implemented in marking m, and generate its follow-up marking;

5) Operate the removal process of the redundancy marking to delete some marking in open;

6) For each follow-up marking m' of each m:

Calculate $f(m')$;

If m' is not in open table and close table, then add it into open table;

If m' is in the open table or close table, then compare the valuation function value of the newly generated $f(m')$ and the previous m'. If the old value is small, then to delete the new generated marking m'; if the new value is small, then to replace the old value, and make the pointer to a new path. If the matched m' is in the close table, then move it into open table;

7) Turn to 3).

The removal process of redundant marking:

1) If open table is empty, then quit;

2) If ti is the change that can be triggered in marking m, then:

Find the sister marking mb of m in open table, namely there is the same father marking between mb and m;

If mb is generated by father marking through ti, then delete mb in open table;

Repeat the above two steps till the entire sister markings of m are checked in open table.

3) Repeat step 2) till all the change that can be triggered are checked in marking m.

The above removal process of redundant marking is based on the principle of change concurrent in Figure [1]. If changes in t1 and t2 are concurrent, they are triggered in arbitrary order, however, only one can represent this situation, the rest can be deleted.

4.Case Studies

In this paper, the final test session of an actual IC production was scheduled. The problems can be abstracted as follows[10]: There are 4 workpieces, and the processing amount of each workpiece is different, for each piece need have three processes, there is only one machine can operate in each process at one time. The description of system's production processes is shown in Table 1.

TABLE I: JOB REQUIREMENT OF EXAMPLE

operation/ workpiece	J1	J2	J3	J4
1	(M1，2)	(M3，4)	(M1，3)	(M2，3)
2	(M2，3)	(M1，2)	(M3，5)	(M3，4)
3	(M3，4)	(M2，2)	(M2，3)	(M1，3)

In Petri net model, pool represents an operation or a state of resources, change represents the start and end of an event or operation process, the end of change for one behavior means starting for the other behavior. Token in operating pool means it is operating. The time interval between start and end operation is the execution time of operation. Thus the system Petri net model is established, as is shown in Figure 1.

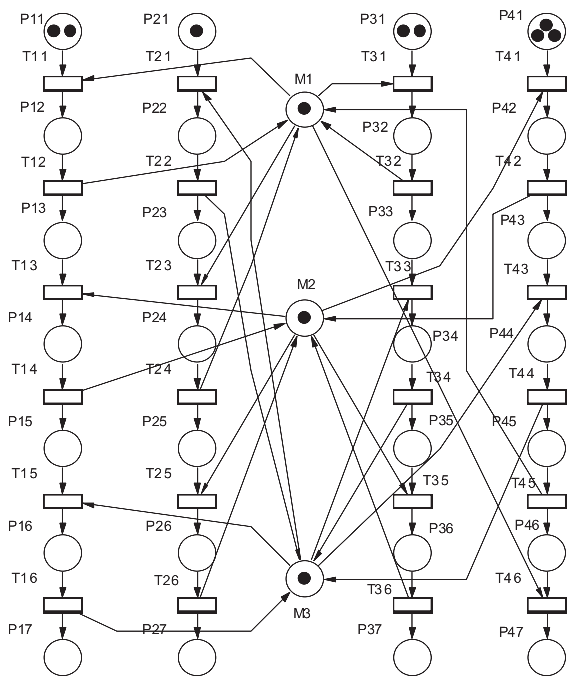

Fig. 1 Petri net model of the example system

The algorithm proposed in this paper makes scheduling for system model under different production batches, and BF algorithm and BT-BF algorithm are compared with (the results are shown in Table 2). It can be seen that scheduling optimization can be realized through this algorithm (H-HS), greatly reducing the search space (it can be reflected from the number of search marking), saving the computer memory, and increasing computing speed.

TABLE II: COMPARISON OF THE PROPOSED ALGORITHM AND OTHER APPROACH

Batch				Makespan			Number of search marking		
J1	J2	J3	J4	BF *	BT-BF *	H-HS **	BF *	BT-BF *	H-HS **
5	5	2	2	58	62	59	3437	1687	813
8	8	4	4	100	104	103	9438	8045	1889
10	10	6	6	134	148	137	23093	18875	2657

* is the computing result of [10], ** is the computing result of the algorithm in this paper.

The scheduling results are listed by using the algorithm under the larger production volume, as is shown in Table 3. It can be seen that algorithm can better overcome the previous algorithm which can not be used to schedule larger system.

TABLE III: SOME LARGE CASES

Batch				Makespan	Number of search marking	CPU time* (second)
J1	J2	J3	J4			
20	20	10	10	253	5520	323

50	50	20	20	583	13817	1858
100	100	100	100	1703	35532	15883

* is the CPU operating time in PC.

5.Conclusions

A hybrid heuristic algorithm that solves the FMS system scheduling problems based on Petri model is proposed in this paper. The dynamic search window technique is combined with the best first algorithm and retrospective methods, and the search space is further reduced by using the removal process of redundancy marking. According to the scheduling optimization of the final testing process for actual IC production, especially to calculate the scheduling results of each workpiece under different batches, and to compare with the computing results of previous literatures, it can be seen that the search space is greatly reduced, the computer memory is saved and the computing speed is increased by this algorithm based on the guarantee of scheduling global algorithm. This algorithm has the characteristics of strong generality, so it can be also used for the scheduling problems of larger system.

References

[1] JENG M D，Chen S C · Heuristic Search Based on Petri Net Structure for FMS Scheduling[J] · IEEE Transactions on Industry Application，1999，35（1）：196-202

[2] RODAMMER F · A Recent Survey of Production Scheduling[J] · IEEE Transactions System, Man and Cybernetic，1988，18（6）：841-851

[3] LEE D Y，DICESARE F · Scheduling Flexible Manufacturing Systems Using Petri Nets and Heuristic Search[J] · IEEE Transactions on Robotics and Automation，1994，10（2）：123-132

[4] SHIH H，SEKIGUCHI T · A Timed Petri and Beam Search Based on-line FMS Scheduling System with Routing Flexbility[C] · Proceeding of the 1991 IEEE International Conference on Robotics and Automation，Sacramento，CA，1991，2548-2553

[5] WANG Xiao-rong, WU Tie-jun. Flexible Jobshop Scheduling Based on Petri-net Model:ACO-GA Hierarchical Evolutionary Optimization Approach[J]. Journal of Zhejiang University(Engineering Science) ，2004，38（3）：286-291

[6] ZHANG Weijun, Freiheit T, YANG Huashu. Dynamic Scheduling in Flexible Assembly Based on Timed Petri Nets Model[J]. Robotics and Computer-Integrated Manufacturing, 2005, 21:550-558

[7] XIE Nan, LI Ai-ping. Reconfigurable Production Line Modeling and Scheduling Using Petri Nets and Genetic Algorithm[J]. Chinese Journal of Mechanical Engineering ，2006，19（3）：362-367

[8] PAN Quan-ke, ZUO Feng-chao, ZHU Jian-ying. Petri Net Based Model for Jobshop Scheduling Problem in Green Manufacturing and Its Hybrid Metaheuristic Search[J]. Chinese Journal of Mechanical Engineering ，2006，42（9）：48-53

[9] SUN T H，CHEN C W and FU L C · A Petri Net Based Approach to Modeling and Scheduling for an FMS and a Case Study[J]. IEEE Transactions on Industrial Electronics，1994，41（6）：593-601

[10] XIONG H H, ，ZHOU M C · Scheduling of Semiconductor Test Facility via Petri Nets and Hybrid Heuristic Search[J] · IEEE Transactions on Semiconductor Manufacturing，1998，11（3）：384-393

[11] JENG M D，CHEN S C · A Heuristic Search Approach Using Approximate Solutions to Petri Net State Equation for Scheduling Flexible Manufacturing Systems[J] . The International Journal of Flexible Manufacturing Systems，1998，10：139-162

[12] MORO A R，YU H and KELLEHER G · Advanced Scheduling Methodologies for Flexible Manufacturing Systems Using Petri Nets and Heuristic Search[C] · Proceeding of the 2000 IEEE International Conference on Robotics & Automation，SanFrancisco，CA，2000，2398-2403

[13] GU Tian-long, XIONG Tai-ping, ZHAO Ling-zhong. Novel Symbolic OBDD Algorithm for Production Scheduling in FMS[J]. Computer Integrated Manufacturing Systems，2007，13（2）：338-344

[14] 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