Journal of Convergence Information Technology
Volume 5, Number 10. December 2010

Dynamic Integrated Algorithm for Production Scheduling Based on Iterative Search

[1,2]Song Cunli, [1]Liu Xiaobing,[1]Wang Wei,[2]Huang Ming
1 *Dalian University of Technology, Dalian 116024 , P.R.China*
**2 Software institute, Dalian jiaotong University, Dalian 116052, P.R.China*
Correspondent: Song Cunli, E-mail: scunli@163.com
doi:10.4156/jcit.vol5. issue10.20

Abstract

It is quite difficult to adapt to the changing production scheduling problem for an algorithm of a certain question. According to this, a strategy of dynamic integrated iterative search algorithm is proposed. Firstly, the relationship model of production problem and algorithm is built. Secondly, the framework of the iterative search algorithm is proposed. On the basis of these, the strategy of dynamic integrated iterative search algorithm based on multi agent is put forward, which can easily adapt to the changeable workshop production environment. The experimental results show the effectiveness and adaptability of the system.

Keywords: *Dynamic Integrated Algorithm, Iterative Search, Multi Agent, Production Scheduling*

1. Introduction

According to the research of product scheduling algorithm, hybrid algorithm has become a hotspot, its basic thought is to design an efficient, robust and optimal algorithm through integrating different methods. For example, by combining the local search, it can improve the performance of the algorithm. For most intelligent algorithm, such as genetic algorithm, evolution strategy, evolutionary programming, ant colony optimization, particle swarm optimization and so on, neighborhood search operators has a significant effect on the performance and efficiency. Therefore many domestic and foreign scholars have proposed many hybrid algorithms. For example, based on an expert system, Zhou Guohua [1] proposed a strategy that dynamically mixed genetic algorithm and other algorithm, and discussed the intelligent expert system's structure and implementation mechanism ; Liang Xu etc. [2] has mixed ants' local search with genetic algorithm to improve the algorithm convergence and speed. Chai Yongsheng [3] has combined the memory function and quick search ability of the immune algorithm with genetic algorithm to improve the quality and speed. However, these hybrid algorithms are all for specific problems, hybrid algorithm universal mechanism and the general representation is not proposed, so are static integrate algorithm. These hybrid algorithms are difficult to adapt to a dynamic production environment. About the theory of hybrid algorithm, Li and Jiang [4]has proved the global asymptotic convergence of mixing the GA, SA and trends algorithm. Krasnogor and Smith [5] has proposed the elements of hybrid algorithm and there relationship model, and given some guiding principle to design the effective hybrid algorithm. WangLing, LiuBo [6] has put forward the hybrid optimization algorithm of swarm intelligence unified description, and analyzed the performance of hybrid intelligent optimization algorithm. But these studies only stay in theoretical studying and no specific realization strategies is proposed.

In this paper, A strategy on dynamic integrate iterative search algorithm is put forward. Firstly, the relationship model of production problem and algorithm is proposed. Secondly, the frame of dynamic integrate iterative search algorithm is proposed. On the basis of this, the strategy of dynamically integrated algorithm based on multi agent is put forward, which made the algorithm can easily adapt to the changeable product environment. The experimental results show the effectiveness and adaptability of the system.

2. Problem description

Production scheduling problem is to plan the production, such as the processing sequence, the lot sizing ect. It can be described as how the M machines process N Jobs to achieve the satisfactory results and satisfy the constraints.

The problems should be solved in this paper are:

a. The description of the production scheduling problem. There are many classes of production scheduling problems, according the flow pattern, Maccarthy has divided the problem as Job shop, Flow shop ect. The problem is how to use a universal module to describe different product scheduling problem.

b. At the base of studying the problem and algorithm, the frame of self-adaptive iterative search algorithm should be proposed. The frame is the base of dynamically integrate, at the support of it, can we quickly generate the hybrid algorithm to solve the problem according to the problem's class, feature, objective.

c. The description of knowledge is another problem. It is the foundation of dynamically integrate algorithm and mainly describe the relationship between scheduling problem with algorithm and the relationship of algorithms.

3. Idea of dynamic integration algorithm

3.1. Production scheduling model

For production scheduling problem, when we design an algorithm, generally should consider the number of task, machines, Processing constraints and goals, that is we should first module the problem. Here, we can adopt Conway's [7] four descriptor system modeling to module the problem, that is the A/B/C/D description method.

A - number of tasks, the possible value for A is positive integer, it often with n.

B - number of machine, it may be a positive integer, it often with m.

C - task flowing form, technique and management constraints ,the possible value are as follows

/ /　: named null values, expressing single workshop,

J: expressing Job-shop(multitasking workshop)

F: Flow-process workshop

F,perm: Replacement flow-process workshop

k-parallel: K machine parallel

j，k-parallel: Each processing stage has k machine parallel multitasking workshop

D- performance targets, specific reference [6].

3.2. Algorithm model

There are many algorithms for the production scheduling problem, but each one has it's own feature. Analytical and enumeration method can solve small-scale scheduling problems easily but different for large-scale problem. The constructive[17] method may quickly solve the problem but the quality is usually poor. The iterative search algorithms such as genetic algorithms, ant algorithm, particle swarm optimization, evolutionary programming, immune algorithm ect. , often can find the problem's optimal solution form a single solution or multiple solutions by iteratively searching , and are the most widely used scheduling algorithms. Therefore the dynamic integrate iterative search algorithm is highlighted in this paper. Eien and Smith [8] proposed the general framework for iterative search algorithm that include six algorithm elements. On the basis of this the frame of the iterative search algorithm is proposed in this paper as follows:

```
Initial module
While (not matching the stop rule) do
    Social cooperation
    Self-modify
    Neighborhood search
    Elimination mechanism
EndW
```

The initial module complete the initialization of the individuals, it can produce an initial individual, matching the single iteration algorithm (taboo search etc.). It may also produce a number of individuals, matching the parallel iterative algorithm (genetic algorithm ect.).

The social cooperation module represents the information exchange and mutual learning behavior of individual in the optimization process, but the iterative algorithm does not have this operation.

The self-modify module represents that the individual actively or passively adjust its status, so as to enhance its adaptability.

The neighborhood search algorithm is added to iterative search algorithm in order to speed up the convergence, expand the search area and enhance the intelligence of algorithms.

The elimination mechanism is corresponds to selection modules and evaluation modules, it guarantees the algorithm can get a better solution.

It is not enough to only know the framework of adaptive iterative search algorithm from the perspective of dynamic integrate, we must describe the module in the database for future intelligently dynamic integrate algorithms. Therefore it is need to study how to describe the algorithm in the database. According to the need some properties of algorithms are proposed in this paper are as follows:

Name: the name of the algorithm

Encoding rules: such as: coding based on the workpiece, Coding based on process ect.

Category: Functions of the algorithm, the concrete classify is as follows:

①Initial module (expressed by the code 1)

②Appraised module(expressed by the code 2)

③Neighborhood search module（expressed by the code 3）

④self-improve module (expressed by the code 4)

⑤Collaboration Module (expressed by the code 5)

⑥selection module (expressed by the code 6, the Elimination mechanism is consisted by the appraised module and the selection module)

Special Interface: it is used to describe the information of special parameters needed by the module. In order to facilitate the communication between algorithms and improve the flexibility of the integrate algorithm, the common interface information needed for the most of the algorithm is replaced by public variables.

Performance targets: it's the basic index to chose the appraise module.

Position: describe where the algorithm placed.

Characters: it is used to describe the problems solved by the algorithm. Such as Job shop scheduling, Flow shop scheduling.

Scale: it is used described by the value range of N*M.

Whether using directly or not: the value is yes or not.

Mixture weights: it is used to state the probability of the mixture between this algorithm and other algorithm. Once success, it adds 1, the initial value is 0.

Outside the database, the common interface to facilitate the communication between algorithms and improve the flexibility of the mix algorithm should be described, in this paper use public variables and symbols to realize the problem, such as:

N: it expresses the number of task

M: it expresses the number of machine

A[N+1][M+1]: it expresses the time of each task producing on each machine

B[N+1][M+1]: it expresses the limits of the task producing on the device

W_{ij} :the time of task i waiting for process j

W_i: the total time of task i waiting

S_{ij}: the time of task i starting process

C_{ij}: the time of task i ending process

L_i: the time of task i lateness

T_i:the time of task i completing tardiness

E_i:the time of task i pre-completing

I_i:the time of machine mi rest

NW(t)：the number of the part waiting for producing at the time of t

Nc(t)：the numberof the part completed process

3.3 Knowledge Representation

Knowledge is the rule of selecting the relevant algorithm to combine on the base of existing task characteristics and requirements, in accordance with the framework Iterative search algorithm , can described an knowledge as follows:

The type of problem: it describes the type of the task and is used to indicate the solved problem.

Scheduling Objective: it describes the goal of the schedule

Initial module: it is corresponding to the frame of iterative research algorithm

Social cooperation module: it is corresponding to the frame of iterative research algorithm

Self-modify module: it is corresponding to the frame of iterative research algorithm

Neighborhood search module: it is corresponding to the frame of iterative research algorithm

Appraisal function: the elimination mechanism is consisted by the appraised module and the select module

Select module :it included by elimination mechanism

Appropriate scale: (small, middle, big of three scales)

Effect: it describes the characters of algorithm program.

The probability of searching excellence: using the times of searching excellence or the times of algorithm executing.

Mean square deviation: it is used to describe the stability of the algorithm, solving the mean square deviation based on the goal of dispatcher.

There are two way to acquire knowledge. One way is to research the existing international and national scheduling algorithm, write the optimal algorithm to algorithm table directly and the knowledge of them to knowledge table. The Second way is schedule the typical production scheduling problem and analysis the resolution of it with the optimal resolution, write the good result to knowledge table.

4. The strategy of dynamically integrated algorithm based on multi agent

In this paper, the multi-Agent system structure diagram is as the Figure.1, each problem Agent's duties is to receive production tasks, according to production tasks, check the production process information, if the task type can match with its own, then the task information is sent to according algorithm agent immediately by message system. The algorithm agent's duty is to dynamic generate integration algorithm or calling the current algorithm to schedule the problem. The main function of learning agent is to receive messages (algorithm mixed strategy) sent by the algorithm agent, analysis and compare the performance of the integration algorithm , if the result is better then write it to the knowledge table.

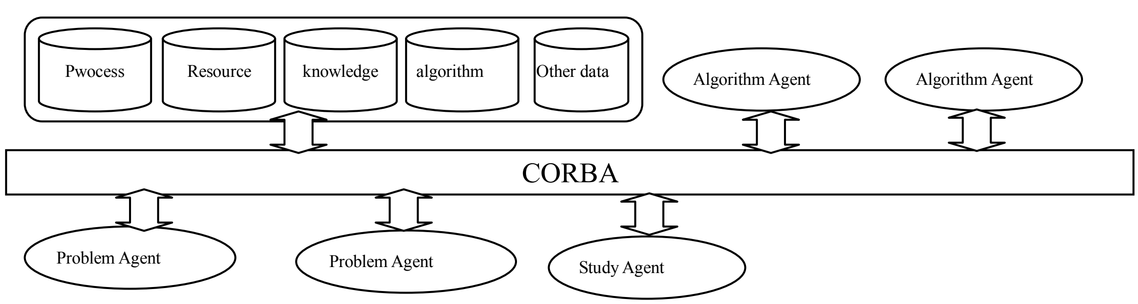

Figure 1. Multi-Agent system architecture diagram

The realization and the key technology should be referred to the literature [9]. Because of the complexity of algorithm agent ,so the it is highlighted in this paper.

Algorithm is the most complex of all agents. Firstly, it receives messages sent by the problem agent, and matches them. If pattern matches successfully, then it detects whether the current task algorithm exists in the knowledge base or not. If it exists, it calls the corresponding algorithm to computer, and then decides the order of the task. If not, it searches the algorithm table, determines the initial module, collaboration module, self-improvement module, neighborhood search, evaluation modules and selection module according to the turntable gambling probability, and integrate an iterative algorithm, then the integrated algorithm calculates the task 10 times, output the best result T and records the integrated program. As the performance of the algorithm is greatly impacted by the neighborhood search, self-improvement and collaboration module, so the neighborhood search module is written to the taboo table, and replaced by another neighborhood search algorithm according to the turntable gambling probability. The best result T* is outputted after running 10 times, compared with T. If it is better than T, then T is changed to T* and the integrated program is modified as well, otherwise the T is not to change, continuing the neighborhood replace until using out the times of alternative neighborhood, here the times the neighborhood alternative is set $\alpha=5$. The same approach can be used to modify the collaboration module and self-improvement module. Finally, the best scheduling results are output and the best integrate programs are written to the knowledge base, the weight of each module used in the integrate program should be increased to improve the probability of module being chosen. in the future.

The pseudo code of algorithm agent description is as follows:

```
Begin:
  While (detecting the problem agent sends information)
      If (the problem agent matches with the task agent) then
          To find the appropriate knowledge base matching the goals
        If (find) then
              To computer directly and schedule the task.
        Else
              The performance of the algorithm is greatly impacted by the neighborhood search,
            self-improvement and collaboration module, the neighborhood search module is
            written to the taboo table, and replaced by the other neighborhood search algorithm
            according to the probability. The best result T* is outputted after running 10 times,
            compared with T. If it is better than T, then T is changed to T* and the integrated
            program is modified as well.
              N=1;
              While (N<=α)
                  To choose the other neighborhood search module, replacing the current
                neighborhood module according to the dial gambling probability.
              If (T* is better than T) then
                T=T*;
                To modify the integrate program;
              End if
                  N=N+1;
            End w
                Replace the social-collaboration module and self-adaptation module in the
              same way.
                Increase the corresponding weight of integrate program.
            End if
         End if
    End w
End
```

5. Simulation analysis

The system is implemented in VC++, experimented on the processor of P（R52）M1.86GHZ and 512MB RAM. The parameters of related algorithm are fixed value, for example the population size is 40 or 1, the probability of social collaboration is 0.8, the probability of self-improvement is 0.05 and maximum iterative algebraic is 200. In the process of dynamic integration, the replacement time of the neighborhood module, collaboration module and self-improvement is 3, 3 and 5 respectively. In the algorithm library, there are total 50 modules are collected such as the modules of initialization, collaboration, self-improvement and so on. Three typical production models are considered (namely job shop, flow shop and no wait flow shop), the targets is to minimize the makespan, every dynamic integrate algorithm will run 10 times. For the job shop problem, three typical problems of fisher and Thompson[10] will be solved, and compare the result of the integrated algorithm with the reference[3] and [11], the results is shown in table 1.

Table 1. the result of Job-shop problem

Problem		reference[11]			reference [3]			Algorithm in this paper			
n	m	C_{best}	P_{best}	Dev	C_{best}	P_{best}	Dev	C_{best}	P_{best}	Dev	time
6	6	55	80%	0.09%	55	70%	0.23	55	80%	0.11%	2.35h
10	10	930	60%	0.26%	930	30%	1.24	930	50%	0.36%	4.23m
20	5	1165	70%	0.38%	1165	40%	1.70	1165	60%	0.58%	3.65m

n is the number of job, m is the number of machine, Cbest is optimal solution, Pbest is the probability of optimal solution, Dev is mean of deviation , time is the running time of algorithm

The formulation of deviation is (1) and (2):

$$Dev = (solution - Optimal\ solution) \Big/ optimal\ solution \tag{1}$$

$$Dev = \sum_{i=1}^{10} Dev_i \Big/ 10 \tag{2}$$

For flow-shop problem, three typical Taillard problems will be solved, and compare the result of the dynamic integrate algorithm of this paper with reference [12] and [13], the results is shown in table 2.

Table 2. The result of Flow-Shop problem

problem	scale		reference [13]			reference [12]			Algorithm in this paper			
TA	n	m	Cbest	Pbest	Dev	Cbest	Pbest	Dev	Cbest	Pbest	Dev	time
Ta005	20	5	1250	40%	0.56%	1235	60%	0.38%	1227	80%	0.12%	1.20h
Ta050	50	10	3189	20%	1.34%	3099	30%	1.60%	3126	60%	0.79%	2.15h
TaA080	100	10	5953	30%	0.88%	5864	50%	0.81%	5834	60%	0.46%	2.45h

For the problem of No wait Flow-Shop, three typical REC scheduling problems will be solved, compare the result of the dynamic integrate algorithm of this paper with the reference [14-15], the results is shown in table 3.

Table 3 . The result of no wait Flow-shop problem

problem	**scale**		**reference[15]**			**reference [14]**			**Algorithm in this paper**			
REC	**n**	**m**	**Cbest**	**Pbest**	**Dev**	**Cbest**	**Pbest**	**Var**	**Cbest**	**Pbest**	**Dev**	**time**
REC07	20	5	2074	80%	0.56%	2074	80%	0.79%	2089	100%	0	1.16h
REC31	50	10	4622	40%	1.57%	4643	50%	1.13%	4676	80%	0.13%	2.05h
REC37	75	20	8461	30%	1.46%	8671	40%	1.25%	8488	70%	0.33%	3.45h
REC39	75	20	8860	40%	1.14%	8878	60%	0.96%	8902	70%	0.24%	3.36m

Scale of the scheduling problem here, used to divide as the small-scale problems when n * m <= 100, the medium size when 100 <n * m <= 500 , the large-scale problems when n * m> 500. For the job shop problem, from table 1 can draw the conclusion that the integrated algorithm can get the optimal solution, however, its probability of optimum is relatively lower than reference [11], higher than reference [3]. The mean deviation of the 10 times is lower than reference [3], It takes 2.35 hours to run when sole the problem of 6*6, which is higher than reference [11] and [3]. But when solve the last two problems, the running time is only 3-4 minutes and the three algorithms are nearly equal. The reason is that the scale of the last two problem is same with the first problem, so the algorithm agent do not spend time to integrate algorithm, it directly use the knowledge of the first problem, so it run fast.

Considering the flow-shop problem form the table 2, can draw the conclusion that the integrated algorithm are better in the optimal solution, optimization rate and in the average deviation than the given reference. But for the running time, because the scale of the three problem is different each other, the running time of the integrated algorithm of the three problem is all more than 1 hours and longer than the given algorithm.

Considering the no wait flow-shop problem form the table 3, that integrated algorithms are slightly worse in seeking optimal solution than reference[14-15]. But the optimization rate and the average deviation are better than the reference[14-15]. Similarly, the running time of integrated algorithms in this paper of the three problem takes a long time because the different scale of the problem, But it only takes 3.36m to solve REC39 problem as there is a knowledge has been written to the knowledge table while solve the problem REC37.

Table 4. statistical data on 3 Algorithm for

	First algorithm	**Second algorithm**	**Third algorithm**
Times of occurrence	3	1	1
Rate of occurrence	60%	20%	20%
Optimal solution	3380	3391	3383
deviation	0.0%	0.3%	0. 08%

To demonstrate the effectiveness of the strategy of dynamic integration algorithm, the same no-wait flow-shop REC29 problem is resolved 5 times, at each time the knowledge table is emptied, the objective is to minimize the makespan, each integrated algorithm is executed 10 times and the optimal solution is recorded. Form table 4 the best integrated algorithm appears 3 times, the occurrence rate is 60% and these prove the effectiveness of the strategy. Compare the three optimal solutions of the three integrated algorithm, the biggest deviation is only 0.3%, and this also proves the robustness of the strategy.

6. Conclusion

At first, this paper analyzed the problem that should be solved for dynamically integrate algorithm. Then the relationship model between production problems and the model between algorithms is proposed respectively. At last, the framework of iterative search algorithm and strategy of dynamically

Integrate algorithm based on multi-agent is put forward. The simulation has proved the effectiveness of the strategy. The strategy can make the scheduling system quickly adapt to the changing production patterns, improves the resilience and competitiveness of the enterprise. The effectiveness of the strategy has been proven by compared the integrated algorithm with the other algorithms of the reference on the same scheduling problem. The disadvantage is that the algorithm's parameters are set manually not automatically. We should continue research on how to set the algorithm parameters adaptively, how to divide algorithm module size rationally to reduce the complexity of algorithms such as dynamic integration.

7. Acknowledgment

The work is supported by the national natural Science foundation, china (no. 70572098), the liaoning provincial educational foundation, china(no.L2010086).

8. References

[1] Zhou Guohua. *Research on Soft Computing for Production scheduling problems*. PhD Thesis of Southwest Jiaotong University:72-79
[2] LIANG Xu, WANG Jia, HUANG Ming.New coding method for massive production scheduling problem[J] . *Computer Integrated Manufacturing Systems*.2008,14(8): 1566-1570
[3] CHAI Yongsheng, SUN Shudong,YU Jiangjun,WU xiuli.Job Shop Dynamic Scheduling Problem Based on Immune Genetic Algorithm[J] .*Chinese Journal of Mechanical Engineering*. 2005,41(10):23-27
[4] Li B, Jiang W. A novel stochastic optimization algorithm. *IEEE Trans. Syst*. Man. Cy.B.:Cybernetics 2000,30:193-198
[5] Krasnogor N, Smith J.A tutorial for competent memetic algorithms: model, taxonomy , and design issues. *IEEE Trans. Evolut. Comput.* ,2005,9:474-488
[6] WANF Ling, LIU Bo. *Particle swarm optimization and scheduling algorithms[M]*. Tsinghua University Press. 2008.5:15-33
[7] Conway, R..W., Maxwel, W.L. and Miller, L.w..*Theory of Sheduling*. AddisonWesley, Reading, Mass, 1967
[8] Eiben A, Smith J E. *Introduction to Evolutionary Computing*. Heidelberg: Springer,2003
[9] XIE Junyi, TANG Wencheng,NI Zhonghua. Research on scheduling management system for manufacturing resources in shops based on multi-agent system. *Computer Integrated Manufacturing Systems*. 2005, 11(6):805-809
[10] Muth J F, *Thompson G L. Industrial Scheduling[M]*.New Jersey:prentice-hail,1963
[11] Chang Guijuan,ZHANG jihui.Immune genetic algorithm based on orthogonal experiment for scheduling problems. *Chinese Journal of Information and Control*. 37(1):46-51
[12] Zhang Changsheng, SUN jigui, YANG Qingyun, ZHENG Lihui. A hybrid algorithm for Flowshop scheduling problem. *Acta Automatica sinica*. 2009.35(3):332-336
[13] Nearchou A C. Flow-shop sequencing using hybrid simulated annealing[j]. *Journal of intelligent Manufacturing*,2004,15(3):317-328
[14] Grabowski J, Pempera J. Some local search algorithms for No-wait flow shop problem with makespan criterion. *Computers &Operations Research*,2005,32(8):2197-2122
[15] Pan Quan-ke, ZHAO Bao-hua, QU Yu-gui, Heuristics for the No-Wait Flow Shop Problem with Makespan Criterion[J], *Chinese Journal of Computers*, 2008,31(7),1147-1154