Proceedings of the 6th World Congress on Intelligent Control
and Automation, June 21 - 23, 2006, Dalian, China

A Hybrid Strategy Based on Ant Colony and Taboo Search Algorithms for Fuzzy Job Shop Scheduling *

Xiaoyu Song [1,2], Yunlong Zhu [1], Chaowan Yin[1], Fuming Li [1,2]

1) Shenyang Institute of Automation , Chinese Academy of Sciences, Shenyang 110016, China

2) Graduate School of the Chinese Academy of Sciences, Beijing 100039, China

Abstract - A hybrid strategy base on ant colony and taboo search algorithms is proposed for fuzzy Job Shop scheduling propose, which uses the ant colony algorithm as a global search algorithm, and adopt taboo search algorithms as a local search algorithm. TS algorithms have stronger ability of the local search, which can overcome the disadvantages of ant colony algorithms, so this hybrid strategy can improve the quality of solutions. The experimental results show that the proposed hybrid algorithm has gotten higher the average agreement index than that of taboo search algorithms and parallel genetic algorithms in solving the hard benchmark problems.

Index Terms **-** Ant Colony algorithm, Taboo Search algorithm, Hybrid algorithm, Fuzzy processing time.

I. INTRODUCTION

Job-shop scheduling problems are well-known as one of the combinatorial optimization problems.

Ant colony algorithms use sufficiently the positive feedback mechanism of ant colony's behavior to get solutions. The algorithm can find the solutions in multi-point of the global by the parallel calculation. Dorigo et al. have applied ant colony algorithm to many classical optimization problems, such as TSP, and obtained the good results. Stutzle and Hoose proposed MMAS system [1]. Ant colony algorithms have successfully applied to flow shop scheduling and group shop problems[2-4]. Colorni et al. firstly used ant colony algorithms to solve job-shop problem [5], but the results is common[4], which is determined by shortcomings of MMAS.

MMAS and other ant colony algorithms have the following disadvantages: First, the positive feedback mechanism of ant colony algorithms can not make the algorithm escape from a local optimum. Second, the searching time of ant colony algorithms is too long. Especially, when the initial pheromone is scarce, large amount of time is needed. TS algorithm is one of the best methods which are used to solve the job-shop scheduling problem. Furthermore, the investigations show that TS provides better guidance than other algorithms in the whole search field[6]. But TS algorithm also has the following shortcomings. Firstly, the quality of solution highly depends on the iterations of algorithms. Secondly, the optimal result relies on the quality of the initial solution. Thus, we can choose some solutions initial solutions of TS algorithm in ant colony algorithm, then use TS algorithm to search the solutions in parallel in the small fields. When the initial pheromone is scarce, the result of TS will offer pheromone effectively. Furthermore, TS can make the search scope larger, which is in favor of the ant algorithm to avoid falling into a local optimum. At last, owing to the fast finding solution of TS, the total time that the algorithm costs is reduced.

In this paper, the fuzzy number [7] is adopted in the fitness function for the fuzzy Job Shop scheduling. We propose a hybrid strategy base on ant colony and taboo search algorithms for fuzzy Job Shop scheduling, which uses the ant colony algorithm as a global search algorithm, and adopt taboo search algorithms as a local search algorithm. TS algorithms have stronger ability of the local search, which can overcome the disadvantages of ant colony algorithms, so this hybrid strategy can improve the quality of solutions. The experimental results show that the proposed hybrid algorithm has gotten higher the average agreement index than that of taboo search algorithms and parallel genetic algorithms in solving the hard benchmark problems.

II. PROBLEM DESCRIPTIONS AND MODEL

An $n\times m$ job-shop scheduling problem can be formulated as follows:

1) Suppose n jobs processed on m machine M_r(r=1, 2…, m), and every job includes a set which is made up of many operations and a fuzzy due date with respect to client's agreement degree.

2) The processing sequence of jobs is defined in advance, and each operation of a job has its own processing time.

3) Under the limited time, one machine can only process one job and each job can be processed in one machine.

4) There is no limitation for the processing sequence of different job, and each operation can't be interrupted. The final goal is to find a feasible solution determining the processing sequence of operations in every machine which is compatible with the craft in the case of that the scheduling is achieved and the due date of a job is guaranteed.

Because of the effect of some uncertain factors on the procedure of processing, we only obtain a approximate range of processing time on a job, so we denote the processing time with triangular number $\tilde{P}_{ij}(p_{ij}^1, p_{ij}^2, p_{ij}^3)$,where $\mu_{ij}(x)$ is used to evaluate the probability of fuzzy processing time, and its

* Supported by the National Natural Science Foundation of China (70431003), the National Basic Research Program of China (2002CB312204).

1-4244-0332-4/06/$20.00 ©2006 IEEE

distributing function is defined by Formula 1, and where x denotes the fuzzy processing time.

(1)

A fuzzy due date is a time window related with the agreement index, which can be represented by trapezoid fuzzy number $\tilde{D}_i(d_i^c, d_i^a, d_i^b, d_i^d)$. We hope that the value of a fuzzy due date is between $[d_i^a, d_i^b]$ but in $[d_i^c, d_i^a]$ or $[d_i^b, d_i^d]$. We use $\mu_i(c)$ to represent the probability of the fuzzy processing time as a function of the trapezoid fuzzy number, which the probability distributing is defined by Formula (2), where c denotes the value of possibility.

$$\mu_i(c)=\begin{cases}1 & c\in\left[d_i^a, d_i^b\right]\\ (c-d_i^c)/(d_i^a-d_i^c) & c\in\left[d_i^c, d_i^a\right]\\ (d_i^d-c)/(d_i^d-d_i^b) & c\in\left[d_i^b, d_i^d\right]\\ 0 & c< d_i^c, c> d_i^d\end{cases} \quad (2)$$

As both the fuzzy processing time and the fuzzy due date are fuzzy, we can only use the agreement index of customers for the optimal goal.

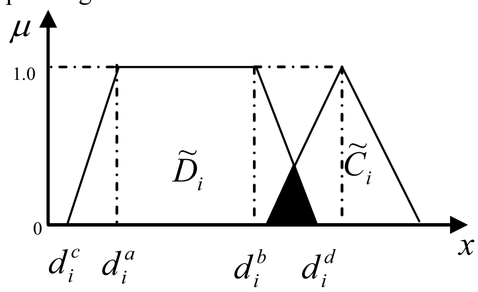

Fig.1 Agreement index of Customers

Definition I. Agreement index. The agreement index named AI is defined as the ratio between the intersection area encompassed by the functions which are subject to the fuzzy completion time and fuzzy due date respectively and the area encompassed by the fuzzy completion time function, which is represented as Formula (3), where $\tilde{C}_i$, $\tilde{D}_i$ denote the area surrounded by the fuzzy completion time and fuzzy due date respectively, and Π denotes the set of feasible scheduling.

$$AI=(\text{area}\,\tilde{C}_i\cap\tilde{D}_i)/(\text{area}\,\tilde{C}_i) \quad (3)$$

For a given scheduling solution $\sigma\in\Pi$, $f(\sigma)$ is defined as the objective function, equal to $min\ \{AI_i(\sigma)\mid i=1, 2, \ldots, n;\ \sigma\in\Pi\}$, where i is the number of a job. Model of max-min agreement index is described as Formula 4, where σ^* is the optimum order which maximizes the minimum satisfaction.

$$f(\sigma^*)=\max_{\sigma\in\Pi} f(\sigma)=$$

$$\max\{\ min\ \{AI_i(\sigma)\mid i=1, 2, \ldots, n.\ \ \sigma\in\Pi\}\ \} \quad (4)$$

III. Solutions of Model

A. Hybrid strategy of algorithms

This hybrid algorithm named TSANT is shown in Figure 2. In the figure, the left part denotes the frame of an ant colony algorithm, and the right part includes some parallel TS algorithms for the local search. After a new swarm is created, TS algorithm optimizes the swarm, and the result as the final swarm.

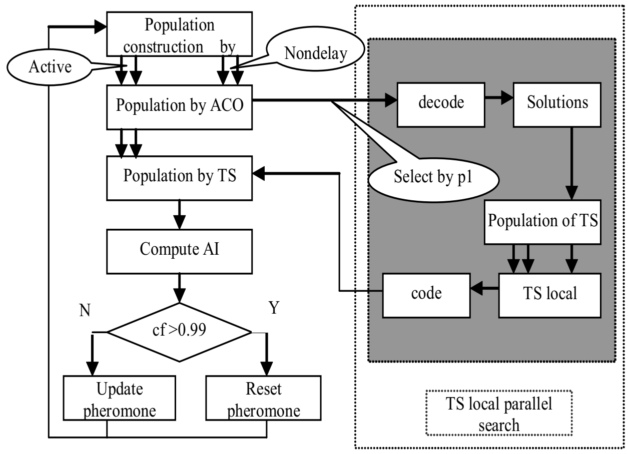

Fig. 2 The framework of TSANT searching strategy

When the ant colony algorithm evolves after some generations, all values of pheromone in the information list are close to τ_{max} or τ_{min}, so the algorithm will fall into a local optimum. Therefore, the converging factor cf is computed after each iteration. If cf is larger than 0.999, we restart the algorithm, that is to say, reset the pheromone value. Thus, the searching ability of algorithm can be enhanced and the early-maturing problem also can be solved. The value of cf can be calculated by Formula 5.

$$cf=2\cdot\left(\left(\frac{\sum_{\tau_{ij}\in\tau}\max\{\tau_{\max}-\tau_{ij}, \tau_{ij}-\tau_{\min}\}}{|\tau|\cdot(\tau_{\max}-\tau_{\min})}\right)-0.5\right) \quad (5)$$

Algorithm 1
Input: A fuzzy job-shop scheduling problem P.
Output: the searched optimal solution
Begin
  S_{bs}=NULL, S_{rb}=NULL, S_{ib}=NULL, iter=0, cf=0, P_l=0.2,
  N=DetermineNumberOfAnts(P) //initialize the population
  InitializePheromoneValues(τ) //initialize pheromone
  while do not satisfy the end condition **do**
    for j = 1 to n **do**
      ConstructionSolution () // construct solutions
    end do
    ApplyLocalSearch() // apply local search
    ParallelTabooSearch(Pl) // parallel TS search

$S_{ib} = max\{AI(S) \mid S \in iter\}$ //S_{ib} the optimal solution
Update(S_{ib}, S_{bs}, S_{rb}) // S_{rb} optimal solution after restarting
ApplyPherononeUpdate(τ ,S_{rb}) // update pheromone
cf = ComputeConvergenceFactor(τ)
if cf > 0.99 **then**
ResetPheromoneValues(τ) // reset pheromone
S_{rb} = NULL
end if
iter++
end while
End

B. Design of the ant colony algorithm

To construct the solution of the ant colony algorithm, we improve the G&T algorithm as follows.

1) Let $Q(1)=\{o_{ij} \mid i=1,\ldots$ and $j=1,\ldots,m\}$, which is a set of all operations, $S(1)$ is a set of all jobs' first operations.
2) Let t=1.
3) Let o^* be a operation which satisfies $c(o^*)=\min\{c(o_{ij} \mid o_{ij} \in S(t))\}$, where m^* is the machine. Select a operation o_{im^*} from $\{o_{im^*} \in S(t), r(o_{im^*})<c(o^*)\}$.
4) Generate $Q(t+1)=Q(t)\backslash\{o_{im^*}\}$. Delete o_{im^*} according to $s(t)$, and add the next operation of job to consist of set $s(t+1)$.

$$P(o_i \mid \tau) = \frac{\min_{o_j \in G} \tau_{ij}}{\sum_{o_k \in G} \min_{o_j \in G} \tau_{kj}} \qquad (6)$$

Different from selecting an operation from the contradictory assemble randomly in traditional G&T algorithm, we chose operations from the contradictory set by the probability denoted by Formula 6.

5) If $Q(t+1)$ is null, let $t=t+1$, turn to step3, else stop.

We call operations o_i and o_j in the same machine the related operations. τ_{ij} denotes the pheromone when o_i is processed before o_i. τ_{ij}' denotes the value of updated pheromone. We solve S_{ib} to updated pheromone denoted by Formula 7.

$$\tau_{ij}' = \tau_{ij} + \rho \cdot (\delta(o_i, o_j) - \tau_{ij}) \qquad (7)$$

,where $\delta(o_i, o_j) = \begin{cases} 1000 \text{ if } o_i \text{ is processed before } o_i \\ 0 \text{ if } o_i \text{ is processed before } o_i \end{cases}$

$$\tau_{ij} = \begin{cases} \tau_{\min} & \text{if } \tau_{ij} < \tau_{\min} \\ \tau_{\max} & \text{if } \tau_{ij} > \tau_{\max} \\ \tau_{ij} & \text{if } \tau_{\min} < \tau_{ij} < \tau_{\max} \end{cases}$$

C. Neighbour exchanging strategy

The critical path is the longest path without time intervals between operations in an available scheduling. For example, in Figure 3, there are two critical paths. The first one is (4,1) (5,1) (3,2) (2,4) (2,5) (3,4) (3,5) (6,4) (6,5) (1,6), and another one is (4,1) (5,1) (5,2) (6,1) (3,3) (1,2) (1,3) (3,5) (6,4) (6,5) (1,6).

We can decompose the operations in the critical path into blocks. A block is a set of neighbourhood operation in a critical path on a machine. For example, the operations (3,5) and (6,4) in the first critical path consist of a block.

When critical path is not unique, not all the neighbour exchanges of operations can turn the critical path short. Figure 3 shows the gantt of 6×6 problem, exchanging operation (2, 5) and operation (3, 4) do not shorten the length of critical path, because St(3,5)= MAX(Et(1,3),Et(3,4)). Although the operation (3, 4) is ahead, the operation (3,5) depends on the operation (1,3) and the operation (3,4), the length of critical path is not reduced. That is to say, some neighbor exchanges are meaningless.

Definition II. The critical operation. Let $J_p(v)$ and $J_n(v)$ represent the previous operation and the next operation of operation v in job, and $M_p(v)$ and $M_n(v)$ denote the previous operation and the next operation of operation v in the machine, $S_t(v)$ and $E_t(v)$ are the start time and the end time of the processing respectively. If $S_t(v)$= $Et(M_p(v))=Et(J_p(v))$ in critical path, then v is called critical operation.

For example, in figure 3 operation (3,5) is a critical operation, the exchanges of previous operations of v is unable to shorten the critical path.

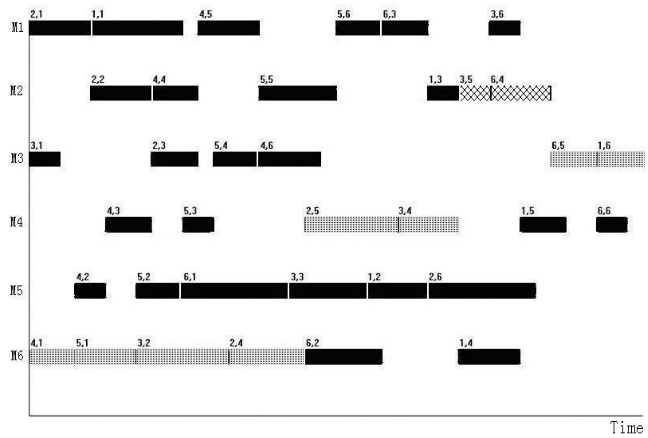

Fig.3 Gantt graph for the solution for FT06 problem

The method of choosing neighbours based on the critical operations is as follows. If there is no the critical operations in the critical path, we can select the exchanged neighbours, if there are critical operations, we can select an exchanging operation between the last critical operation and the last operation. Thus, many inefficient searches are pruned. If TS algorithm can not find the improved solution after 50 steps, the search is stopped.

IV. EXPERIMENT EVALUATION

The experiment environment is: OS Window XP, Memory 512M, CPU 2.4GHz, and C language.

We use triangular number $\widetilde{t}_{ilk}$ ($t^1_{ilk}, t^2_{ilk}, t^3_{ilk}$) to denote the processing time of experimental job, where t^2_{ilk} uses 13 typical hard benchmarks, including FT10, LA02, LA19, LA21, LA24, LA25, LA27, LA29, LA36, LA37, LA38, LA39, and LA40, and then fuzzed. Moreover, $t^1_{ilk} = t^2_{ilk} -$ rand(0, t^2_{ilk} ×0.2), $t^3_{ilk} = t^2_{ilk}$ +rand(0, t^2_{ilk} ×0.2).

The experimental parameters are as follows. The ant population is |O|×30% in TSANT. The evolutionary generations are |O|. The probability of using TS is 0.2, and after 50 steps of TS, if the solution does not get better then terminates it. ρ is random number between 0 and 1, in the experiment, ρ = 0.1, and τ_{min} = 10, τ_{max} = 1000. All initial pheromones are 500. The amount of all PGA swarm are 200, all the evolutions are done for 20 generations, the crossed probability is 0.85, and permuted probability is 0.05. TS algorithm is TSAB in [8], and after 3000 steps of searching procedure, if the solution is still not improved, and then TS stops.

TABLE I
The optimal value and average of ten experiments results

Problems	Average of agreement index			Time
	TSANT	PGA	TSAB	(sec.)
FT10(10×10)	0.835	0.815	0.803	60.35
LA02(10×5)	0.740	0.724	0.729	15.37
LA19(10×10)	0.879	0.856	0.867	59.25
LA21(15×10)	0.865	0.832	0.826	198.26
LA24(15×10)	0.769	0.737	0.708	206.58
LA25(15×10)	0.821	0.806	0.785	210.36
LA27(20×10)	0.784	0.721	0.725	298.36
LA29(20×10)	0.687	0.682	0.692	288.37
LA36(15×15)	0.805	0.837	0.806	369.15
LA37(15×15)	0.889	0.852	0.838	359.58
LA38(15×15)	0.873	0.839	0.848	375.16
LA39(15×15)	0.906	0.871	0.863	364.12
LA40(15×15)	0.773	0.734	0.726	376.48

As shown by Table 1, the agreement index obtained from TSANT rises 3.11 percent compared with PGA, and rises 4.01 percent compared with TSAB. For the last seven benchmarks, with respect to the average of ten agreement indexes, TSANT is worse than TSAB only on LA29, and worse than PGA and TSAB on LA36. While summing up the seven problems, the average agreement index rises 3.27 percent and 3.98 percent respectively. It fully demonstrates that embedding TS in ant colony algorithm provides searching information for TSANT effectively, and enlarges the scope of searching, then increases the searching availability. Even thought the quality of solutions searched by PGA and TSAB is well, the average value obtained by TSANT hybrid algorithm achieves further improvement, which verifies the validity of hybrid ant colony algorithm. The right column of Table 1 is running time of different algorithms, comparing with other algorithms, TSANT cost shorter time.

V CONCLUSIONS

The traditional single algorithm can't solve effectively the fuzzy Job Shop scheduling. This paper embeds TS algorithm in ant colony algorithms successfully for the job-shop problem, which provides searching pheromone effectively, magnifying the scope of searching, and improving the searching efficiency. The experimental results on 13 hard benchmarks problems show that the proposed hybrid algorithm is effective, and it provides a feasible and effective approach to the ant colony algorithm for fuzzy Job Shop scheduling.

REFERENCES

[1] Thomas Stutzle, Holger H. Hoose. MAX-MIN Ant System [J]. Future Generation Computer Systems. 2000, 16(8): 889-914.

[2] Chandrasekharan Rajendran, Hans Ziegler Ant-colony algorithms for permutation flowshop scheduling to minimize makespan/total flowtime of jobs [J]. European Journal of Operational Research 2004, 155: 426–438.

[3] Kuo-Ching Ying, Ching-Jong Liao. An ant colony system for permutation flow-shop sequencing [J]. Computers & Operations Research 2004, 31: 791–801.

[4] Christian Blum, Michael Sampels. An ant colony optimization algorithm for shop scheduling problem [J]. Journal of Mathematical Modelling and Algorithms. 2004, 3(3): 285-308.

[5] Colorni A, Dorigo M, Maniezzo V, Trubian M. Ant system for job-shop scheduling [J]. Belgian Journal of Operations Research, Statistics and Computer Science 1994,34(1):39–53.

[6] Jain.A.S, Meeran.S. Deterministic job-shop scheduling: Past, present and future [J]. European Journal of Operational Research 1999, 113: 390-434.

[7] Masatoshi Sakawa,Ryo Kubota. Fuzzy programming for multiobjective job shop scheduling with fuzzy processing time and fuzzy duedate through genetic algorithms [J]. European Journal of Operational Research 2000, 120: 393-407.

[8] Eugeniusz Nowicki, Czeslaw Smutnicki. A Fast Taboo Search Algorithm for the Job Shop Problem [J]. Management Science, 1996, 42 (6): 797- 813.