

Scheduling on Flexible Exhibition Logistics Based on Genetic Algorithms

Zhigang Chen^{1, a}, Tao Yang^{2, b}, Hehua Li^{3, c}

¹²³Shanghai Second Polytechnic University, Shanghai, China

^azgchen@edu.sspu.cn, ^btaoyang@ems.sspu.cn, ^chhli@ems.sspu.cn

Keywords: Flexible exhibition logistics; Job-shop scheduling; Genetic algorithm

Abstract. The exhibition logistics, which generally deal with set-up, breakdown, transport, and storage to ensure that all specific exhibition requirements are met. It is very special from traditional logistics service that its transportation distant is very short and transportation cost can almost be ignored. So the optimization of exhibition logistics is similar to job shop scheduling. In this article, The framework of exhibition logistics is illustrated by a job-shop schedule modal with a kind of flexible on machine selection. Secondly we used genetic algorithm (GA) to deal with problem of job shop scheduling. Finally the experiment result indicate that the proposed algorithm is feasible and effective for this problem.

Introduction

The exhibition logistics, which generally deals with set-up, breakdown, transport, and storage to ensure that all specific exhibition requirements are met, are gained more attentions than before. Generally, exhibition or trading affairs consist of the delivery of exhibits from the door of the exhibitors to the sites of exhibition on time, and delivery of the equipments and materials from exhibition place to the other sites designated by exhibitors at the end of the exhibition, event or show. By the aspect of process, exhibition logistics usually made up of four phases: firstly freight forwarding at exhibitors' place, secondly pre-exhibition logistic service, thirdly on-site logistics service, lastly post-exhibition logistic service. The freight forwarding at exhibitors' place often provided the service of packing and collection of goods from exhibitors' place, FOB formalities, custom clearance and ocean or air freight. The pre exhibition logistics provided the service of movement from entry point to exhibition site, offloading the goods and intermediate storage. The on-site logistics included unloading at site, setup of stand, delivery to stands, unpacking and assembly, removal and storage of empty cases. The post exhibition logistics included breakdown of the stands, removal from stands, dismantling and repacking, movement of exhibits to a nominated address or storage depot ready for next exhibition.

The first phase of exhibition which mostly deal with freight forwarding with almost on difference with ordinary logistics. And the other phases including pre and post exhibition services as well as on-site services, is very special from traditional logistics service. In these phases the transportation distant is very short and often constrain to a small areas close to the exhibitions venues, So the transportation cost of these phases can almost be ignored.

The optimization problems of exhibitions logistics is similar to job-shop schedule because of its ignorable transportation cost and its emphasis on the effective use of the limited resources. Job shop scheduling is an optimization problem to determine an schedule in which ideal jobs are assigned to resources at particular times.

Determining an efficient schedule for the general job shop problem has been the subject of research for more than 50 years. For n jobs and m machines in the general case there will be $(n!)m$ feasible sequences[1]. Scheduling problems are known to be complex even for simple formulations and are NP-hard in many cases. The automatic generation of scheduling plans for job shops are traditionally addressed using optimization and approximation approaches [2]. Optimization algorithms include enumerative procedures and mathematical programming techniques consisting of

linear, integer programming, goal programming, transportation, network and dynamic programming. The techniques included in approximation techniques are: implicit enumeration (branch-bound); decomposition (Lagrangian relaxation); priority rule based; heuristics; local search algorithms (iterative search, simulated annealing, threshold annealing, Tabu search); evolutionary programs (genetic algorithms); AI techniques (knowledge-based and expert systems). Optimization algorithms provide satisfactory or optimal results if the problems to be solved are not too large and are restricted to low dimensional over simplified problems. Approximation algorithms are problem dependent and raise the problem of overcoming the difficulties between computer and human reasoning. In both the cases, the efficiency problem arises[3]. Approximation algorithms are capable of guaranteeing the solution to be within a fixed percentage of the actual optimum and are considered as useful tools for solving discrete optimization problems. Commonly, the genetic algorithms can reach a satisfactory solution to a problem with large scale variables more rapidly. So we choose it to solve the exhibition logistics problem in this article.

Costs Structure of Exhibition Logistics

Generally, the exhibitions logistics providers will handle with the tasks such as setup and breakdown of stands, unpacking and assembly, removal and storage of empty cases, movement to or removal from stands, dismantling and repacking, intermediate storage, surrounding the exhibitions venues as specified by customers. These tasks will be operated with the limited resources of all kinds of facilities including machines, tools and entrance of inspection etc. Thus the total costs include the cost of usages of these facilities, the switch cost of a specified facilities when switching from one job to the others, the storage costs when the exhibits is in a state of waiting and being stored in intermediate depots surrounding or within the exhibition venues, and the late costs for no being finished on time when the events begin..

Problem Description and Modeling

For the research purpose, we uniformly called various kinds of exhibition logistics service designated by the exhibitors as jobs, whatever it was setup of stands, unpacking and assembly, or removal and storage of empty cases, etc. Meanwhile all facilities necessary for handling the exhibition logistics, whatever they were machines, tools or entrances, etc, were uniformly called machines for convenience.

The exhibition logistics provider was given certain jobs from various customers, which need to be scheduled on certain machines. Each job consists of some operations which being processed over machines. The operations of each job are processed with a given set of machines with predefined sequence, while not necessarily the same set of machines. Some assumptions were summarized as follow:

- Each job consists of several operations with a predefined sequence.
- Each operation can be processed on one machine out of a set of given machines with predefined processing time.
- Once started, operations cannot be interrupted.
- Each machine can only process one job at a time.
- the operations of the same job must be processed with predefined sequences.
- Job cancels and machine breakdowns are not allowed.

Mostly, job shop scheduling problem concerned with trying to minimize the make-span[4]. The make-span is the total length of the schedule when all the jobs have finished processing. But for exhibition logistics, things are different. The customers will require all the operations are being done before exhibition day. Therefore the exhibition logistics providers would try to minimize the total cost while still satisfying the customers' time constraints. So the objective function can be to minimize the total cost of exhibition logistics providers.

Supposed the exhibition logistics provider was given n jobs $J_1, J_2 \dots J_n$ of various customers, which need to be scheduled on machines $M_1, M_2 \dots M_m$. Each job consists of several operations, with the j th operation of the i th job denoted as O_{ij} . Operation O_{ij} can be processed by a machines set denoted as M_{ij} . The processing time of operation O_{ij} over machine M_k is denoted as P_{ijk} .

Let $[E_i, L_i]$ be the time window of job i , E_i and L_i , respectively be the earliest completion time and the latest completion time of job i . C_i is the actual completion time of job i . Let A_i be the penalty coefficient for job i completed ahead of the predefined earliest completion time E_i , and D_i is the penalty coefficient for delay completion of job i .

Our object is to minimize the total cost for the logistics provider which mostly consisted of the storage costs for each job and the delay cost for failure to complete jobs before the dates specified by the customer. Thus the object function is to minimize the storage cost and idle cost for all job.

$$TC = \text{Min} \sum_{i=1}^n [D_i \times \max(C_i - L_i, 0) + A_i \times \max(E_i - C_i, 0)]$$

The total costs for logistics provider is TC , including the storage costs for exhibitions items and delay costs for each jobs of customer. The storage cost is calculated by penalty coefficient for completion ahead of the predefined earliest time. The storage cost was proportional to the time that exhibition items are at rest when it needs to occupy the depots or warehouses of the exhibition places. The delay cost is calculated by penalty coefficient for delay completion. The delay cost was the penalty for the jobs no being completed on time. Usually, the delay cost per unit time was set to a large number to make sure that all jobs would be completed before the exhibition date.

Genetic Algorithms

Genetic algorithm (GA) is a search heuristic that mimics the process of natural selection. This heuristic is routinely used to generate useful solutions to optimization and search problems. The exhibition logistics scheduling optimizations is a NP-hard problem. Genetic algorithm is usually used in this kind of problems[5].

Genetic Coding. We use natural number coding to make up the chromosome of the job shop schedule. In this way, we divide the individual solution as two parts. The first part of solution is based on the sequence of operation, while the second part of solution is based on machine selection with which one machine is chosen from feasible machine sets for each operation.

The gene coding base on the operation sequence is presented as a serial of natural number such as (1,2,3,2,1,3,3,2) for the operations sequence of ($O_{11}, O_{21}, O_{31}, O_{22}, O_{12}, O_{32}, O_{33}, O_{23}$) in the Table1. The gene coding based on the machine selection is also composed of a serial of natural numbers such as (1,1,2,3,2,3,2,1) in the Table1. It means the machine numbers for operations ($O_{11}, O_{21}, O_{31}, O_{22}, O_{12}, O_{32}, O_{33}, O_{23}$) are (1,1,2,5,3,5,2,1).

Job	operation	Processing Time				
		M1	M2	M3	M4	M5
Job1	O11	3	4		5	
	O12		2	4		
Job2	O21	3		3	2	5
	O22		4	2		1
	O23	2	3		1	3
Job3	O31	1	4	5	2	1
	O32	4			3	2
	O33	3	1	3		

Table1: Operation Sequence, Machine Sets And Processing Time

Crossover. Crossover is a genetic operator used to vary the programming of a chromosome or chromosomes from one generation to the next..

Crossover based on operation sequence: In order to ensure the feasibility of the offspring schedules, we restrict the crossover position within all operations of the same job which is randomly selected. The chromosome of offspring1 inherits the corresponding genes from the positions chosen from parent1, with the rest of genes filled in the same order, from the remaining genes of parent2 with which the chosen genes of parent1 is deleted. The offspring2 is designed with the similar way. It is clear that crossover in this way will not produce invalid genes. The crossover is showed as the Table2 with the job1 chosen in parent1 and job2 in parent2.

Parent1	2 1 2 4 1 3 4 2 1 3 3 4
Parent2	2 2 4 1 3 2 4 3 3 1 1 4
Offspring1	2 1 2 4 1 3 2 4 1 3 3 4
Offspring2	2 2 1 4 1 2 3 4 1 3 3 4

Table2: Crossover Based On Operations

Crossover based on machines selection: First, a binary set composed of 0,1 is randomly generated with the length equal to the chromosome. Then, the genes of two parents, corresponding to the 0 of the binary set, swap and thus generate two offspring. In order to ensure the feasibility of the offspring schedules, we replaced the gene with the one whose processing time is shortest from the feasible machine set if the original gene is infeasible. The crossover based on machines selection is as Table3.

Random	1 0 1 0 1 1 0 0 1 0 1 1
Parent1	1 1 3 4 3 3 2 2 1 2 3 4
Parent2	1 2 3 2 3 1 2 3 2 1 2 1
Offspring1	1 2 3 2 3 3 2 3 1 1 3 4
Offspring2	1 1 3 4 3 1 2 2 2 2 2 1

Table3: Crossover Based On Machine Selection

Mutation. Mutation is a genetic operator used to maintain genetic diversity from one generation of a population to the next. Mutation alters one or more gene values in a chromosome from its initial state. Mutation occurs during evolution according to a specified mutation probability. This probability should be set low. If it is set too high, the search will turn into a primitive random search.

Mutation based on operation: We can generate the mutation by randomly selecting an operation from one offspring, for example, the second operation of job1. Due to limitations of the processing sequencing, we exchange its location with another location randomly chosen from the area between the first operation of the job1 and the third operation of job1. Thus it satisfies the processing sequencing limitation.

Mutation based on machine selection: As each operation can be done by more than one machine, so we randomly choose one gene from the chromosome, and replace it with another machine number, also randomly, from the machine set of the specific operation. This can come to a feasible schedule.

Fitness function and selection. During each successive generation, a proportion of the existing population is selected to breed a new generation. Individual solutions are selected through a fitness-based process, where the individual is typically more likely to be selected with a higher fitness.

The fitness function is defined over the genetic representation and measures the quality of the represented solution. In this article the fitness of i^{th} individual in a population is equal to the reciprocal of the objective function value of corresponding solutions in our algorithm.

$$fitness_i = 1/TC_i$$

Selection is the stage of a genetic algorithm in which individual genomes are chosen from a population for later crossover. This article chooses fitness proportionate selection method plus the excellent retention method with which the best one of the population is preserved to the next generation.

Experiment and analysis

The experiment presented in this section is designed to test the effectiveness and performance of our model and algorithm. Now we consider a 6 jobs and 6 machines problem with the operation sequence and the processing time for each operation have been determined in Table4

For convenience, The storage cost per time unit for all job is set to 1. The delay cost per time unit for all job is set to 100. We assume a large number for delay cost in order to make sure that all job will be completed before exhibition date. The time windows for all job is [13,15].

		M1	M2	M3	M4	M5	M6
Job 1	O1 1		3		4	4	
	O1 2		5		3		4
	O1 3	3			1	4	
Job 2	O2 1	4	4		5	6	
	O2 2	4		5		8	
	O2 3		8		7		8
Job 3	O3 1	8	4	5		9	7
	O3 2	2	5	5		2	
	O3 3		5	6		5	
Job 4	O4 1	6	6			6	
	O4 2	3		4	3		3
	O4 3		6	5	8	7	
Job 5	O5 1	8		7		9	
	O5 2	3		2			1
	O5 3		1	1		5	
Job 6	O6 1	6	5		5	5	
	O6 2		5	9		7	
	O6 3		4		1		2

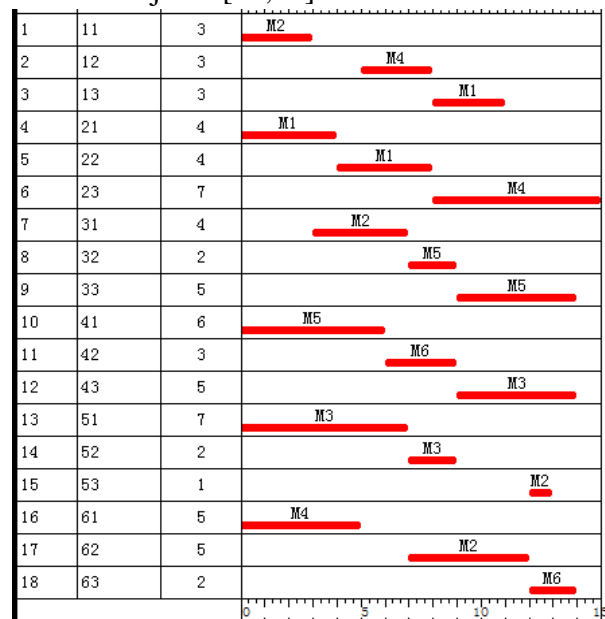


Table4: Data of Jobs and Machines

Fig1. The Result of Scheduling

We use the population size of 10, Probability for mutation is 15%. The algorithm was terminated after 100 generations.

For the result in Figure1, it can be shown that the last job processed is job 2 on machine 4. So, our make-span value for this problem is 15, and the minimal of TC is 2.

The result also gives us the operations sequence and machines selection for each operation, the starting time and the finish time for each operation.

Conclusions

In this article, we conduct cost optimization of the flexible exhibition logistics with machines selection being considered, and provide a job-shop schedule model to optimize its total cost. Then we provide a genetic algorithm to solve the problem. The experimental results show that the algorithm is feasible and efficient for the problem.

Acknowledgment

The research was supported by funding from: Research Foundation of Education Bureau of Shanghai, No.10Y206 and promotion fund of logistics discipline of SSPU .

References

- [1] Davis, Lawrence, Handbook of Genetic Algorithms, Van Nostrand Reinhold, New York, 1991.
- [2] J. H. Holland, Adaptation in natural and artificial systems : an introductory analysis with applications to biology, control, and artificial intelligence. Ann Arbor: University of Michigan Press, 1975.
- [3] T. Yamada and R. Nakano. Genetic Algorithms for Job-shop Scheduling Problems. Proceedings of Modern Heuristic for Decision Support., UNICOM Seminar, 18-19 March 1997
- [4] P. Mellor, A review of job shop scheduling[J], Operation Research Quarterly, 17(2), pp. 161-171, 1966.
- [5] Bingul Z. Adaptive genetic algorithms applied to dynamic multi-objective problems. Applied soft Computing,2007,7.