한국시뮬레이션학회 '99춘계학술대회 논문집 1999.4.17 동의대학교

Optimal Design and Operating Policies in a Direct-Input-Output Automated Manufacturing System Using Simulation Optimization Method

Ji Young Yu and Young Hae Lee

Department of Industrial Engineering,
Hanyang University
jyyu@pis.hanyang.ac.kr

Abstract

We propose a method for simulation optimization using stochastic genetic algorithm(GA) to get the best design and operating policies in a Direct-Input-Output Automated Manufacturing System (DIOMS).

1. Introduction

Most approach for manufacturing system have been studied on either system design problems or operational problems. But, these two problems are interrelated, because the design of the system must be based on the way of the system operation and conversely, the operational problem depends on the design of system. We should consider both system design and operational problems. Suggested method in this paper considers both system design and operational problems using the simulation optimization method. We consider operating policy including input sequence control, dispatching rule for AS/RS, machining center-based part type selection rule, and storage assignment policy and system design including the number of each machine type, machine layout and rack size of AS/RS.

In many simulation optimization methods for manufacturing system design, the layout of the manufacturing system is given and only the set of values for certain parameters of the system is determined. This, however, is not true when seeking optimum designs for most practical systems. The methodology described in this paper is a simulation optimization process where the qualitative variables, the quantitative variables and the layout of the system are optimized.

We propose a method for simulation optimization using stochastic genetic algorithm(GA) to get the best design and operating policies in a DIOMS.

1.1 Direct Input Output Manufacturing System

In recent manufacturing systems design workstations coupled with centralized Work-In-Process(WIP) storage and handling using AS/RS. This system called integrated-automated manufacturing system have proven to be very effective. Through the use of automated Storage/Retrieval(S/R) machines, the storage and handling functions of the production systems have been integrated with the manufacturing operations, thus obtaining better space utilization, real time inventory tracking, better production control, and great flexibility to accommodate process changes[9].

1.2 Simulation Optimization

Simulation optimization can be defined as the process of finding the best input variable values from among all possibilities without explicitly evaluating each possibility. The objective of simulation optimization is to minimize the resources spent while maximizing the information obtained in a simulation experiment[2].

2. Analysis of DIOMS

This system is consist of rack which can store the pallet and a small automatic warehouse which is organized with one stacker crane (S/R machine). Also this system disposes the machining center in one or both side when it is operated. Each machining center selects one among rack openings which are placed in the first floor, bottom of a small automatic warehouse and uses it as a load/unload port[1]. The system is illustrated in Figure 2.1.

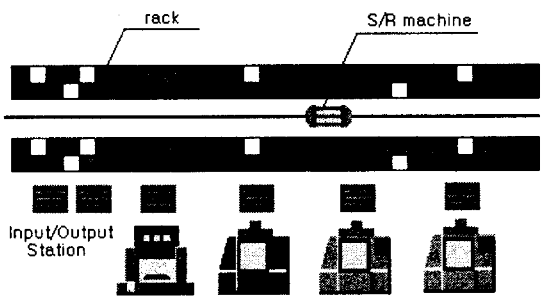

Figure 2.1. Configuration of DIOMS

The operations in DIOMS is performed sequentially as the following manner: jobs enter the DIOMS through I/O station. Material handling equipment such as S/R machine picks up a pallet and places it on the P/D port located in front of a machine. Then the part is automatically fed into the machine. After completion of the assigned operation, the pallet may be moved to another machine for its next operation. If the machine which will perform the next operation is not available, the pallet is moved to the WIP area in AS/RS and stored until the machine becomes available. The pallet which has gone through all the required operations leaves DIOMS via I/O station[9].

The studies on the DIOMS can be classified into two categories, one for the design problems and the other for the operational problems. These two are inter-related, because the design of the system must be based on the way the system operation and conversely, the operational problem depends on the design of system[9]. We consider operating policy including input sequence control, dispatching rule for S/R machine, machining center-based part type selection rule, and storage assignment policy and system design including the number of each machine type, machine layout and rack size of AS/RS.

3. Stochastic Genetic Algorithm for DIOMS

The suggested stochastic genetic algorithm(GA) to get the best design and operating policies in a DIOMS has two step. The first step in the method is to get the set of good alternatives using GA and simulation. The second step is to get the best solution of m alternatives obtained in the first step using the screen-and-selection method.

3.1. GA and simulation

3.1.1 Representation

In order to supply GA for DIOMS, chromosome is configured as following :

chromosome
= *{substring1, substring2, substring3, ... , substring n }*
= *{ {a1,a2,a3,a4}, {b1,b2,b3,b4}, {c1,c2,c3,c4}, ... , {n1,n2,n3, ... , nk} }*

A string consists of several substring. Each substring in a string represents the decision alternatives regarding one aspect of the DIOMS._

3.1.2. Evaluation of Objective function

Our Objective is to minimize system cost per unit time while there is no constrains. The cost include machine cost, AS/RS cost and throughput cost. Machine cost and AS/RS cost are setup cost and throughput cost is operation cost by the unit time. So objective function can't contain machine cost and AS/RS cost directly. Machine cost and AS/RS cost are modified to cost of unit time. C_i is the cost of ith machine per unit time. C_i is calculated by machine setup cost per unit time added to maintenance cost per unit time. The machine cost is the sum of cost of all machine by the unit time. It's as following:

$$\sum_{all\ i} n_i C_i$$

The basic structure of AS/RS is one level. If one level is not enough, we have to add one or more level. The cost of basic structure of AS/RS($C_{AS/RS}$) include land cost, setup cost etc. Therefor, it is difference $C_{AS/RS}$ and cost of additional one level of AS/RS(C_{add}) . Total cost of AS/RS is $C_{AS/RS}$ added to cost of additional levels(n_{level}). $C_{AS/RS}$ and C_{add} are AS/RS cost per unit time also. It's as following:

$$C_{AS/RS} + n_{level} C_{add}$$

The DIOMS is required to work upon only objective throughput. If observed throughput(n_{obs}) is short or over target throughput(n_{obj}), penalty cost ($C_{penalty}$) is added to system cost. Object function is as following:

$$\min \sum_{all\ i} n_i C_i + C_{AS/RS} + n_{level} C_{add} + C_{penalty} \left| n_{obj} - n_{obs} \right|$$

3.1.3. Selection

Chromosomes from the current population are selected with a given probability and copies from these individuals are created to constitute the mating pool. Selection of individuals is based on their fitness relative to the current population, i.e., the strongest individuals will have a higher probability of being in the mating pool. Fitness is determined by the objective function that we wish to optimize.

In a stochastic simulation it is not possible to conclusively rank any population of solution without expending excessive simulation effort (number of replications). Since we cannot rank the solution with certainty, we expend just enough simulation effort to divide them into a minimum number of distinct groups of solutions. We then assign the same selection probability to all solution within a group, specifically the average selection probability across all of the ranks in the group[7]. We divide solutions with Subset Selection Approach[8].

3.1.4. Crossover

Crossover is the mating process allowing for information exchange. Crossover method used in Stochastic GA is one-cut-point method, which randomly selected one cut-point and exchanges the right parts of two parents to generate offspring. Figure 3.1 shows how one-point crossover creates two new individuals through swapping all the bits beyond the randomly-chosen crossover point.

Each substring in a chromosome performs crossover. Therefore, one chromosome performs three crossovers. This method is very efficient for some case that a chromosome have varied variables. Because feasible solution retain after crossover. Each substring affect fitness value each other differently so that crossover rate is applied to obtain rapidly optimal solution.

example p1 = { 4,2,2,4, 4,2,1,3, 1,2,5 }
p2 = { 1,2,4,3, 2,1,3,4, 1,3,4 }

p'1 = { 4,2,4,3, 4,1,3,2, 1,2,5 }
p'2 = { 1,2,2,4, 2,4,1,3 1,3,4 }

3.1.5 Mutation

The mutation operator randomly chooses a mutation point with a low probability, and flips the bit at that point. Mutation operator is affected by the type of variable.

The definition of the mutated gene b'_{pos} is dependent on the type of gene that is at position *pos*. The three following cases can be occurred.

1) b_{pos} is an integer gene as substring1. We select the new value b'_{pos} in the interval D_{pos} =[min, max], according to a uniform distribution. It is possible either to use a discrete distribution (e.g., binomial distribution with a mode b_{pos}) or to round to the nearest integer the value obtained

crossover point

chromosome A 11111 111111

chromosome B 00000 000000

chromosome A' 11111 000000

chromosome B' 00000 111111

Figure 3.1. One point crossover.

from a continuous distribution (e.g., Gaussian).

2) b_{pos} is an sequence gene as the substring2. we select randomly two genes from the sequence substring and exchange two genes selected. For example, if b_{pos}, b_1 is selected, the chromosome b becomes b'.

$$b' = b_{pos} b_2 \cdots b_1 b_{pos+1} \cdots b_n$$

3) b_{pos} is a qualitative gene as the substring3. In this case we select randomly a new element from the corresponding set D_{pos}. For example, if b_{pos} is the dispatching rule FIFO and D_{pos} ={FIFO,SPT,EDD}, b'_{pos} is selected from the two other rules[7].

3.2 Nelson's screen-search method

We get m solution through GA process. We therefore turn our attention to separating those solutions into the best, near best and inferior solutions. Since we apply only enough error control in the search phase to insure that the search makes progress, it is quite likely that there is too much sampling error in the performance estimates to make these finer distinctions. In this section we describe a screen-and-select procedure that takes the output of our search, eliminates clearly inferior solutions, and obtains enough additional replications to separate the best and near-best from the rest. This procedure is based on [8].

References

[1] Joon Mook Lim, Kil Soo Kim, Ki Seok Sung, Determination of the Optimal Configuration of Operating Policies in an Intergrated-Automated Manufacturing System Using the Taguchi Method and Simulation Experiments, J. of the Korea Institute of Industrial Engineers, 11, 3, 23-39, 1998.

[2] Yolanda Carson, Anu Maria, Simulation Optimization: Methods and Applications, Proceedings of the 1997 Winter Simulation Conference, 118-126, 1997.

[3] Xiaohua Liu, Sushil J. Louis, Combining

Genetic Algorithm and Case-based Reasoning for Structure, Technical Report, University of Nevada-Reno, April 1996

[4] Goldberg, D. E.. Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley, 1989.

[5] Holland, John. Adaptation In Natural and Artificial Systems. The University of Michigan Press, Ann Arbour, 1975.

[6] Farhad Azadivar, George Tompkins, Simulation optimization with qualitative variables and structural model changes: genetic approach, European Journal of Operational Research, 113, 169-182, 1999.

[7] Justin Boesel, Barry L. Nelson, Accounting for Randomness in Heuristic Simulation Optimization, Technical Report, Department of Industrial Engineering and Management Science, Northwestern University, 1998.

[8] David Goldsman, Barry L. Nelson, Statistical Screening, Selection, and Multiple Computer Simulation, Proceedings of the 1998 Winter Simulation Conference, 159-166, 1998.

[9] Joon Mook Lim, Design and Operation Problems in a Flexible Manufacturing System with built in Automated Storage/Retrieval System, Ph.D Thesis, Department of Industrial Engineering, KAIST, 1994.