Scheduling Optimization for Batch Processing Machines Using Advanced Genetic Algorithm

Yimeng Wang
School of Microelectronics, Tianjin University, Tianjin, 300110, China
Corresponding author’s e-mail: ShiLiShuang@cas-harbour.org

Abstract. Semiconductor manufacturing is widely recognized as one of the most complex production processes today, with a great deal of batch processing equipment scheduling problems. At the same time, more and more research and efforts have been made to optimize the process of decision making in scheduling batch processing machines. Based on previous studies that often combine machine learning algorithms with practical production applications, this paper provides a new approach to the scheduling process using an advanced genetic algorithm. Specifically, this paper offers an algorithm concerning both productivity and efficiency, provides the specific steps of the algorithm. Finally, a few possible future directions for the algorithm are discussed.

1. Introduction

In semiconductor production, batch processing equipment scheduling has a significant impact on the production cycle, order delivery level, equipment utilization and other performance indicators. Batch processing equipment refers to the equipment that can process one or more batches of the same product at the same time, meaning the batch number does not interfere with the production time. The number of lots in the batch account for the utilization of processing equipment. It is necessary to make the scheduling decision immediately throughout the working process. Immediate process can reduce waiting time [1]. However, the traditional decision method that starts the process of lot immediately after the arrival of new lot may cause utilization reduction in some circumstances, while prolonging waiting time unwisely apparently increases the production for saving operation cost of the equipment. The clever decision measure is to calculate the time interval between different machines based on the actual practice and make sure the waiting time will not interfere with the productivity in the first place. Nevertheless, not many researches have been done based on this idea using machine learning algorithms.

Nowadays, it is of great theoretical and practical significance to study the job scheduling problem of batch processing equipment. Firstly, considering both the high price of batch processing equipment and the high proportion of the processing time the lot takes on batch equipment in the production cycle, industries hope to improve the equipment utilization rate to reduce the production cycle. Second, due to the batch processing equipment, a group of qualified lots can be processed on the batch equipment at the same time, there is generally a set of combination and sequencing operation before actual processing. Third, in the production of multi-level orders, there is a need to satisfy the priority order processing on the batch processing equipment [2]. Therefore, batch processing scheduling is the key of the production scheduling and how to solve the scheduling problem through new approaches has been a heated topic. Nowadays, artificial intelligence algorithms provide new ideas for solving traditional scheduling problems. By using heuristic iterative algorithms, the optimal solution can be found under specific
problem assumptions. By using this, we can propose a new optimal scheduling system which offers optimal solutions after countless iterations.

In the process of production scheduling and control, heuristic rules are generally used to determine the sequence of batch processing on the equipment. There are many heuristic rules for production scheduling of semiconductor batch processing equipment. They can be mainly divided into three categories: algorithms based on Processing Time, algorithms based on Due Date and algorithms on Waiting Time, among which CR (Critical Ratio), EDD (Earliest Due Date), SPT (Shortest Processing Time), FIFO (First in First out), LS (Least Slack) are commonly used scheduling rules worldwide.

Other than these heuristic rules, strong heuristic rules including ACO (Ant Colony Optimization), GA (Genetic Algorithm), SA (Simulated Annealing), mathematical programming and artificial intelligence algorithms are also commonly used and studied in this field [3]. Glassery et al. proposed a Dynamic Batching Heuristic, in which they take into account the coming the log, by inserting a certain period of idle time for the equipment, which may lead to a poor flexibility. To solve the problem, this paper proposes an advanced genetic algorithm based on the idle time of the equipment in batch processing scheduling, which fills in the blanks in previous researches and also provides a new research idea.

2. Genetic Algorithm

Genetic Algorithm (GA) was first proposed by John Holland in the 1970s, the algorithm was designed according to the evolutionary laws of organisms in nature. It is a computational model of biological evolution process which simulates natural selection and genetic mechanism of Darwin’s biological evolution, and it is a method to search for the optimal solution by simulating the natural evolution process. In this way, the algorithm converts the solving process of the problem into the process of crossover, mutation and choice of chromosome genes and natural choice in biological evolution. Compared with some conventional optimization algorithms, GA can usually get better optimization results faster. Nowadays, GA have been widely used in combinatorial optimization, machine learning, signal processing, adaptive controlling problems and AI.

Genetic algorithm is a random search algorithm that uses biological natural selection and natural genetic mechanism for reference. Its search process is the process of population evolution. It selects the survival of the fittest through the evaluation function and simulates the evolution of organisms through crossover and mutation.

Shuang Bing and Gu Xingsheng studied the scheduling problem of earliest delivery time and last delivery time, designing a mathematical model of continuous process production scheduling based on GA for order time window due date constraints. The feasibility and effectiveness of the proposed model and algorithm were verified by a hypothesis example. Li Ping and Gu Xingsheng study the scheduling problem with minimum advance/delay with common delivery window, introduce the fuzzy operation, establish the fuzzy programming model of the problem, put forward the optimization method based on GA for the clear mathematical model, and verify the effectiveness of the algorithm by the simulation experiment [4].

There are relatively few literatures on advance and delay scheduling of batch processing equipment in semiconductor wafer fabrication under due date. The improved genetic algorithm proposed in this paper considers scheduling optimization under fixed due date.

2.1. Traditional Genetic Algorithm

Gene, in genetic algorithm, is a group of codes, representing the genetic characteristics involved in calculation. While evolution takes place, the population, a group of individuals, their genes are changed in an observable way. Each organism in the Population is an individual, and every individual in the Population is interrelated and affects each other, which affects the evolution of the Population.

The next generation of individuals will inherit part of the genes of the previous generation of individuals, so that the biological characteristics of individuals can be continued to the next generation. However, heredity is not stable. There is a certain probability of gene mutation, and the new biological
characteristics generated by gene mutation may improve or decrease the environmental fitness of individuals. Genetic algorithm can improve the environmental fitness of individuals through fitness evaluation, so that the mutant gene can be carried on to the next generation with greater probability.

From the algorithmic point of view, reproduction is the process of the Crossover algorithm, in which the individuals in the population are interchanged with two parts of the gene coding fragments to get the next generation of individuals. The Mutation algorithm in the genetic algorithm is realized by directly replacing one or several codes in an individual gene. Some algorithms also use to directly generate a new individual, replacing the entire gene code to realize the gene Mutation.

Both gene crossover and gene mutation are important steps of genetic algorithm, but they should not be carried out too frequently, otherwise the gene difference of each generation will be too great, and the algorithm cannot converge to the approximate optimal solution. If the occurrence of gene crossover and gene mutation is too small, it will not work either. The reason is that it cannot guarantee the diversity of the population. Thus, the algorithm may eventually converge to a local optimal solution, so that the global optimal solution cannot be obtained.

Selection is also one of the most important algorithms in genetic algorithms, and it is also the embodiment of the principle of "survival of the fittest" in genetic algorithms. Selection is to select some excellent individuals from the previous generation population to be passed on to the next generation population according to certain rules according to individual fitness. Fitness refers to the degree to which an individual adapts to the environment. Individuals with low Fitness will be gradually eliminated, while individuals with high Fitness will increase in numbers. In the genetic algorithm, a Fitness function is generally set according to the requirements of the problem to evaluate the Fitness of each individual.

Strictly speaking, GA is not a specific algorithm, it represents a kind of thought. For different problems, gene selection and coding, fitness evaluation function design and genetic operator design are different. GA is a typical iteration algorithm, the population from N to N + 1 generation evolution of individuals in the process of evaluation, selection, operation, crossover operation and mutation operation corresponds to the process of biological evolution in natural evolution, breeding, and mutations roles.

2.2. Scheduling Optimization

When GA are used to solve specific problems, the first problem to be solved is the selection and coding of genes. A gene in a genetic algorithm is a solution to a real problem expressed in some form of code, in the form of data that can be stored and processed by a computer. In GA, fitness is used to measure how close individual in the population is to the optimal solution, that is, how good the individual's genes are. Individuals with high fitness are more likely to inherit to the next generation, while individuals with low fitness are less likely to inherit to the next generation. The Function to calculate individual Fitness is Fitness Evaluation Function or Fitness Function. The selection operator in the genetic operator needs to evaluate the probability of each individual inheriting to the next generation according to the Fitness Function of the individual. Therefore, the design of Fitness evaluation Function is always closely related to the selection of genes.

The evaluation process of fitness function for individuals is: Firstly, the genes of individuals in the population are decoded, and the gene coding in the genetic algorithm is converted to the data expression form in the problem space; Secondly, according to the data expression form in the problem space, the objective function or the optimal value evaluation method of the problem space is used to calculate the corresponding results of the problem space. Finally, according to the type of the problem and the form of the optimal solution, the calculated results are evaluated and converted according to certain rules, and the individual fitness of the genetic algorithm is obtained.

Selection operator, crossover operator and mutation operator are known as the three genetic operators of genetic algorithm. Selection operator is the basic way to ensure good gene propagation in genetic algorithm, which corresponds to the phenomenon of population evolution. The crossover operator corresponds to the gene exchange phenomenon generated by species reproduction, while the mutation operator corresponds to gene mutation. The crossover and mutation operators are used to generate new individuals, which is the guarantee of genetic diversity.
The function of the selection operator is to select the individuals that are more suitable to the environment from the population and copy them to the next generation. The basis of the operation of the selection operator is the evaluation value of the individual fitness, so the selection operator and fitness function directly affect the performance of the genetic algorithm [5]. According to the principle of survival of the fittest, the commonly used Selection operators of genetic algorithm are non-uniform Selection. Common Selection strategies include Proportional Selection, Stochastic Tournament, Optimal Reserved Selection, Sorting Selection, Determine Sampling and so on.

The function of crossover operator in genetic algorithm is to exchange part of gene fragments of two individuals to produce two new individuals. To design the algorithm of crossover operator, the general requirement is not to destroy too many good genes in individual genes, but also to produce new individuals with different genes effectively, so as to ensure the diversity of population. The design of the crossover operator is generally determined by the way the gene is encoded. The basic process is to randomly select two individuals from the population to pair, and then exchange the gene fragments in the corresponding positions according to certain crossover rules. The genetic Crossover rules can be roughly divided into single-point Crossover, two-point Crossover, multi-point Crossover, Uniform Crossover and Arithmetic Crossover.

The function of mutation operator in genetic algorithm is to replace the gene fragment corresponding to a certain point on individual gene with the value of other gene fragment suitable for that point, so as to generate a new gene. Like genetic mutation in biological evolution, mutation is only an auxiliary means to generate new individuals in genetic algorithms, and usually a relatively low probability is used to control the frequency of mutation. Mutation operator and crossover operator jointly determine the search performance of genetic algorithm. According to how to determine the location of Mutation and how to carry out gene Mutation, common Mutation operator types were divided into single-point Mutation, fixed-position Mutation, uniform Mutation, Boundary Mutation, and Gaussian Mutation [6].

3. Advanced Genetic Algorithm

In the traditional algorithm, genetic code is expressed meanly by binary code. In the scheduling problem of batch processing machines, the author used time gap between two different process to represent the genetic code. This advanced algorithm focuses on how to minor the time consumption while it can also increase the utilization of machines to some extent. In the design of this algorithm, considering that batch processing equipment has different processing upper limit, different process stages have different processing time, and the actual processing often has requirements on the delivery date of the product. After comprehensive consideration and analysis of these points, it is found that the time interval between two processes is a very important variable, so the algorithm design is based on calculating the time interval to achieve the goal of global production scheduling. The fitness function is based on measuring the time spent of all batches. By using this advanced algorithm, system can make a decision of the waiting time for each batch in the fastest way of production.

The actual production time is calculated with various time intervals in different decisions, which can be used to measure the quality of the decisions. The decision optimization can be achieved through the preliminary screening and multiple iterations of the improved genetic algorithm.

3.1. Batch Process Quantification

Semiconductor manufacturing is still one of the most complex and difficult manufacturing projects at the current level of technology. The manufacturing process mainly consists of four steps: wafer fabrication, pilot test, packaging and finished product testing. Batch processing equipment scheduling plays an important role in semiconductor packaging scheduling and can directly determine the efficiency and quality of semiconductor packaging [7].

The main complex characteristics of batch processing equipment in semiconductor wafer manufacturing are multi-entry processing, complex processing parameters, Multi-product collinear and expensive to manufacture. In addition, the processing time of batch processing equipment is generally longer than that of single-piece processing equipment, the waiting time of the workpiece to be processed
is limited and the work piece to be-processed arrives dynamically. These complex characteristics of semiconductor wafer manufacturing batch processing equipment make its research process complicated [8]. Furthermore, it is difficult to obtain a satisfactory scheduling solution in an ideal time [8].

Considering these actual production scenarios, to design the algorithm for this problem, there need to be some constrictions. For each batch process machine, they need to meet these items.

There is a fixed amount of time for each processing. There is a fixed limit on the number of products that can be processed at a time. Intervals being used to maintain machine effectiveness are not taken into account.

To analysis the scheduling problem with specific constrains mentioned, there is a simplified model: There are three batches material need to be processed by machines, containing 5, 10 and 10 logs respectively. To complete the processing, there are three process machines in total, the time required for which are 3, 2 and 3 minutes respectively while the maximum capacity of the machines are 20, 10 and 15 logs.

First there are some common variables and constants need to be defined:
- \( B_i \) — Number of logs in Batch \( i \)
- \( C_j \) — The maximum capacity of log of process \( j \)
- \( D_{i,j} \) — Waiting time for batch \( i \) before process \( j \)
- \( F_j \) — Time required to finish process \( j \)
- \( ET_{i,j} \) — End time of batch \( i \) for process \( j \)
- \( ST_{i,j} \) — Start time of batch \( i \) for process \( j \)

The total working process can be quantified in time form, corresponding each batch’s start time and end time for each process, and can be calculated using the formula below:

\[
ST_{i,j} = \sum_{j=0}^{j-1} (D_{i,j} + F_j) + D_{i,j}
\]

(1)

\[
ET_{i,j} = \sum_{j=1}^{j} (D_{i,j} + F_j)
\]

(2)

After the time calculation for each batch, the whole working sequence of the line is scheduled. In this case, the time gap between two different processes is essential to the scheduling optimization, which is also the reason to be chosen as gene code.

For this specific simplified model, the optimization result of processing time of each batch is shown as figure 1.

![Figure 1. Process Time of Each Batch in the Simplified Model](image)

Using the wait time before each process as the genetic code, a random number can be used to generate a number within a certain range as the genetic code. The Fitness Function can also be designed according to the total time spent.

3.2. Filter Design and Test

Taking the time that different batches of parts wait before different processes as the genetic code representing the outcome of a decision, there is a need to make sure that the genetic code is consistent.
with the actual production. Therefore, before individuals can enter the evolutionary process, they need to be filtered, and those who meet the following two conditions can enter the population. First, if two batches enter a process at the same time, it is necessary to ensure that the total number of logs they produce does not exceed the limit that the machine can handle. Secondly, if the machine is occupied, the logs of other batches cannot enter production.

In order to ensure the above limiting conditions, the randomly generated genetic codes can be screened and processed by using the following code structure to delete the individuals that do not conform to the actual production.

The designing strategy of the filter is shown in figure 2:

```plaintext
For all timing of all batches
  { if start time of two batches is different
    { if start time of batch i in process m equals end time of batch j in process m
      flag = 1
    } else if start time of two batches is same
    { if two batches' log in sum is not greater than the maximum process capacity
      flag = 1
    }
  }

Figure 2. The Designing Strategy of the Filter
```

Then man can use the computer to generate 100 random individuals with filter. As a result, a part of results is shown in figure 3, including the time spent of different schedules.

```
| Schedules fitting actual practice | The minimum time |
|----------------------------------|------------------|
| [ 9, 10, 16, 8, 1, 16, 14, 5 ] | 43.0             |
| [ 4, 8, 4, 2, 5, 13, 7, 1 ]    | 29.0             |
| [12, 7, 19, 12, 6, 18, 0, 15, 7 ] | 46.0             |
| [ 3, 12, 18, 7, 10, 2, 7, 18 ] | 43.0             |
| [ 2, 0, 16, 20, 17, 16, 8, 13 ] | 61.0             |
| [16, 16, 4, 3, 5, 17, 16, 13, 13 ] | 50.0             |
| [11, 0, 16, 17, 11, 10, 11, 6, 18 ] | 46.0             |
| [16, 18, 0, 16, 2, 14, 20, 4, 2 ] | 42.0             |
```

Figure 3 Result of Random Generation with Filter

With the filter, as much individuals can be generated as the operator wants to begin the GA process. The crossover and mutation operation can be done under a certain probability, the choice of individuals can be made from the Fitness Function based on the time spent. The flow chart of the advanced GA is shown in figure 4:
4. Conclusion and Future Discussion

The improved genetic algorithm discussed in this paper adds a filter based on semiconductor production batch processing equipment on the basis of the original algorithm, and innovates in gene setting and selection operator.

In batch equipment scheduling, the improved genetic algorithm can make the instruction center make the decision based on time spent more quickly, while the utilization of batch machines can also be taken into account in the decision-making process. In the future, production decisions with urgent parts can be made based on this improved algorithm. This heuristic algorithm has high flexibility and effectiveness and can be widely applied to batch machine scheduling problems in semiconductor production and other fields.

In practice, the production line may fail for a variety of reasons, causing production to stop. In this case, the system needs to make new decisions in a timely manner to cope with the changed delivery date and equipment processing capacity. The system not only needs to have the ability of optimal scheduling, but also should be expected to have the ability of real-time decision-making in the future, so as to improve its fault tolerance rate in actual production.

References

[1] Cao, Z., Yu, H., Qiao, F. (2010) Modeling and optimal scheduling of semiconductor production line based on petri nets and genetic algorithm. Acta Electronica Sinica, 38(02): 340-344.

[2] Huang, C. (2012) Semiconductor production scheduling problem based on genetic algorithm. Science and Technology Information, 11: 220.

[3] Li, L., Qiao, F., Jiang, H., Wu, Q. (2004) Research on dynamic scheduling method of semiconductor production line. Computer Integrated Manufacturing Systems, 8: 949-954.
[4] Li, P., Gu, X.S. (2004) Job shop scheduling with different delivery window under uncertain conditions. Journal of Management Science, 7(2): 22-26.

[5] Li, L., Qiao, F., Wu, Q. (2006) Scheduling rules for batch manufacturing equipment in semiconductor production Line. Journal of System Simulation, 9: 2419-2425.

[6] Zhou, X. (2014) Research on batch machining equipment job scheduling problem based on hybrid ant colony algorithm. Fuzhou University.

[7] Xing, Y., Xu, H., Kang, Y. (2019) Batch processing equipment scheduling in semiconductor packaging stage. Electronic Technology and Software Engineering, 16: 119-120.

[8] Jia, W. (2016) Research on batch processing equipment scheduling in semiconductor wafer fabrication. Shanghai Jiaotong University.