行政院國家科學委員會補助專題研究計畫成果報告

※※※※※※※※※※※※※※※※※※※※※※※※※※※※※

※ 一個改進分段處理的排程演算法 ※

※ 及其最佳分段的研究 ※

※ On the improvement of the sectional processing scheduling ※

※ algorithm and the determination of optimal section length ※

※※※※※※※※※※※※※※※※※※※※※※※※※※※※※

計畫類別：☑個別型計畫　□整合型計畫

計畫編號：NSC 90－2213－E－009－116

執行期間：90 年 8 月 1 日至 91 年 7 月 31 日

計畫主持人：林心宇 教授

共同主持人：

本成果報告包括以下應繳交之附件：

□赴國外出差或研習心得報告一份

□赴大陸地區出差或研習心得報告一份

□出席國際學術會議心得報告及發表之論文各一份

□國際合作研究計畫國外研究報告書一份

執行單位：國立交通大學電機與控制工程學系

中 華 民 國 90 年 10 月 31 日

行政院國家科學委員會專題研究計畫成果報告

一個改進分段處理的排程演算法及其最佳分段的研究

On the improvement of the sectional processing scheduling algorithm and the determination of optimal section length

計畫編號：NSC 90-2213-E-009-116

執行期限：90 年 8 月 1 日至 91 年 7 月 31 日

主持人：林心宇教授　交通大學電機與控制工程系

一、中文摘要

半導體產業是台灣經濟發展的一個重量級產業，由於此產業面臨巨大的國際競爭壓力，因此技術的研發、成本的降低、以追求高利潤的營運成績便成為此產業無法忽視的方向。本計劃即針對半導體製程的排程問題提出一個分段處理的的演算法並分析最佳分段的長度。本計畫已完成演算法的模擬及最佳分段的理論分析並進而得到一個移動分段排程演算法。

關鍵詞：半導體製程、排程演算法、最佳化方法、水平退縮控制

Abstract

Semiconductor manufacturing is a very important industrial sector in Taiwan's economy. Because of international competition, semiconductor industry faces high pressure. Therefore, to develop new technology, reduce cost and pursue high revenue becomes an unavoidable direction. Our project propose a sectional processing algorithm to resolve the scheduling problem of a semiconductor fab and analyze a near optimal section length for the sectional processing algorithm. We have completed the simulations and obtain a good result. In addition, we propose a moving section scheduling algorithm that is most suitable for the semiconductor fab.

Keywords: semiconduction manufacturing process, optimization, scheduling.

二、計畫緣由與目的

在半導體製程中，每單位時間的產出量(through put)是利潤的一個重要指標，而為了達到此高利潤的結果，一個良好的排程方法便是不可或缺的工具。一般而言，排程方法大致可分兩個類型，一種是machinwise，一種是 systemwise。對於一個訂單超多的台灣廠商而言，systemwise 的排程法將較具最佳化的意義。然而 systemwise 的排程法的計算時間非常的冗長，主要是由於訂單量的時間長度過長(例如三個月)，及訂單過多等等。因此為了使計算快速起見，分段排程演算法便成為一個較符合實際應用的方法。而更令人驚訝的是，分段排程演算法除了計算時間較快外，它所得到的解比不分段排程法所得到的能更好。於是，我們便進一步分析最佳的分段長度，並希望能藉此分析出的最佳分段長度來提出一個移動分段排程演算法(Moving section scheduling algorithm)。

三、結果與討論

經過嚴謹的理論分析與模擬，本計畫的研究結果大致可以下列三項來說明：

3.1. 分段處理的效果：

Since the length of the horizon and number of operations have been reduced considerably, the dimension of each section's scheduling problem is reduced drastically. Hence, the difficulty of computational complexity is resolved. However, there is an immediate question concerning how much suboptimality will be lost in using this sectional processing algorithm to solve the original scheduling problem. The answer is a surprised one that the suboptimal solution is even better than the one obtained using the original algorithm [2].

In the following, we will analyze this result using qualitative estimation.

Since we arbitrarily decompose the given horizon into consecutive short horizons, the overall optimality cannot be retained unless special technique is designed. Thus, when we apply algorithm M_1 to each section's scheduling problem, the degrade of overall optimality is proportional to the number of sections employed. On the other hand, the scheduling problem is a nonlinear problem, and the list scheduling algorithm M_2, is a linearization technique based on the cost sensitivity with respect to the beginning time. This implies that, algorithm M_2 has a less destructive nonlinear effect over a shorter horizon scheduling problem. Thus, when the number of sections increases, the section's horizon becomes shorter, and algorithm M_2 will achieve a better objective value. The above two effects resulted from M_1 and M_2 due to varying the number of sections are conflicting, therefore, we may possibly determine an optimal parameter, the section length, by empirical simulation results. To demonstrate these qualitative estimation, we test our sectional processing algorithm on some long-horizon scheduling problems in the following section.

3.2. 模擬結果：

Semiconductor manufacturing process is a re-entrant line process where wafers may return many times to the machine for processing at different operations of the process. In addition, fab operations involve multiple product accommodation by a single production line. Each product may consist of several tens or hundreds of operations. Thus, scheduling of this process is a typical long-horizon scheduling problem.

We have tested numerous ling-horizon job-shop scheduling problems which are simplified versions of a semiconductor manufacturing process. Each problem consists of 6 type of products and each production flow consists of approximate 20 operations. There are 11 types of machines, and each type consists of various number of machines ranging form 3 to 15. For the purpose of explanation, we show in Table I the test results.

Table I: Simulation results of the test example

No. of Sections (L)	Objective Value	Normalized CPU times	Section Length $\frac{K}{L}$	Average System times
1	1.0	1.0	250	54.5
2	0.983	0.452	125	
3	0.967	0.545	84	
4	0.966	0.403	62	
5	0.949	0.453	50	
6	0.946	0.469	42	
7	0.965	0.591	36	
8	0.964	0.517	32	
10	0.967	0.517	25	
15	0.947	1.334	17	

The results in the first row represent a set of standard values that our sectional processing algorithm should compare with; therefore, we normalize both the objective values and the CPU times in the first row as 1.0. The time horizon K of

the problem, in this example problem is around 250. Therefore, the section length shown in row 2 and column4 is 125, and so forth. We see that our algorithm achieves not only the reduction of the CPU times but also the reduction of the objective values in almost all cases with different section numbers shown in Table I. These test results verify the concept proposed in this paper that the suboptimal solution is affected by the controlled parameter.

C.移動分段排程演算法：

An interesting fact indicated in Table I is that when the section length, shown in column 4, approximately but not exactly equals the average system time, shown in the last column, the reduction of the CPU times is approximately optimal. In all cases of various section lengths, the objective values are reduced ranging from 2% to 5.4%. The result mentioned above is reasonable in the aspect of the CPU times reduction, because when we decompose the long-horizon scheduling problem into consecutive short-horizon sectional scheduling problems, the CPU times will reduce; however, if the section length is too short, then there will be too many yet complete operations left for next section and cause computational inefficiency. The results regarding system time, the objective value reduction is as we expect. When the section length equals the average system time, the objective value reduction is not the biggest but is among the better ones about 5%.

Nowadays, the foundry receives the orders from numerous fabless IC design houses and results in a long-term fully booked production lines. Consequently the time horizon of the corresponding job-shop scheduling problems is very long comparing with the time unit of the processing time of an operation. Such a long horizon can be considered as infinite horizon. Consequently, we can consider this type of scheduling problem as an infinite-horizon job-shop scheduling problem. Since there are hardly any existing method including our sectional processing algorithm can solve such an infinite-horizon scheduling problem within a reasonable computational time, and even if it can be solved, the obtained schedule cannot accommodate the unexpected events such as unexpected machine failures or the restoration of repaired machines and unexpected urgent orders, etc.. Borrowing the idea from the moving horizon control in model predictive control theory, we propose here a *moving section scheduling algorithm*. Our algorithm is justified by the simulation results of the sectional processing algorithm shown in previous section provided that we choose the section length to be the average system time. Therefore, we can solve the infinite-horizon job-shop scheduling problem by one section at a time using *moving section* scheduling algorithm.

四、計畫成果自評

如前所言本計畫已完成下列工作項目：

(1)分段處理排程演算法的提出與模擬
(2)最佳分段的分析
(3)提出移動分段演算法以解infinite-horizon的排程問題

目前，我們已計畫將所得成果撰寫成期刊論文並擬稿。

五、參考文獻

[1] Wolpert, D.H. and W.G. Macready, "No Free Lunch Throrems for Optimization, " IEEE Transactions on Evolutionary Programming, 1, 67-82, (1997).

[2] Wang J., Luh P., "A combined Lagrangian relaxation and dynamic programming algorithm for job shop scheduling," *Processing of Rensselaer's 5th International Conference on Computer Integrated Manufacturing Automation Technology* Grenable, France, 3-8, (1996).

[3] Ronce E., Arsan T. and Gawthrop P. J., "Open-loop intermittent feedback control: practical continuous-time GPC", *IEE* proceedings, **146**, 426-434, (1999).

[4] Mayne, D. Q., and Michalska, H., "Receding horizon control of nonlinear systems," *IEEE Trans.* Automat. Contr., bf 35, 814-824, (1990).

[5] Cassandras, C. G., *Discrete event systems*, Irwin Inc. and Aksen Associate Inc., (1993).

[6] Luh P., Hoitomt D., Max E., Pattipati K., "Scheduling generation and reconfiguration for parallel machines," *IEEE Trans. Robotics Automat*, **6**, 687-696, De. (1990).

[7] Hoitomt D., Luh P., Max E., Pattipati K., "Scheduling jobs with simple precedence constraints on parallel machines," *IEEE Contr. Syst. Mag.*, **10**, 34-40, (1990).

[8] Chen H., Chu C., and Proth J., "A more efficient Lagrangian relaxation approach to job-shop scheduling problems," Proc. of *IEEE Int. Conf. On Robotics and Automation*, 496-501, (1995).

[9] Bruvold N., Ecans J., "Flexible mixed-integer programming formulations for production scheduling problem," *IIE Trans.*, **17**, 2-7, (1985).

[10] Carlier J., Pinson E., "An algorithm for solving the job-shop problem," *Manag. Sci.*, **35**, 167-176, (1989).

[11] Conternob R., Ho Y., "Order scheduling problem in manufacturing systems," *Int. J. Production Res.*, **26**, 1487-1510, (1988).

[12] Fisher M., "Optimal solution of scheduling problems using Lagrange multipliers, Part I," *Operations Res.*, **21**, 1114-1127, (1993).

[13] Graves S., "A review of production scheduling," *Operations Res.*, **18**, 841-852, (1981).

[14] Wang, X. Z., Data mining and knowledge discovery for processmonitoring and control, Springer, London, (1999).