科技部補助專題研究計畫成果報告
期末報告

台式餐廳求解等待時間變異最小化之出菜排程問題

計 畫 類 別 ： 個別型計畫
計 畫 編 號 ： MOST 105-2410-H-263-003-
執 行 期 間 ： 105年08月01日至106年07月31日
執 行 單 位 ： 致理學校財團法人致理科技大學國際貿易系(科)

計畫主持人： 李政雄

計畫參與人員： 大專生-兼任助理：曾穎萱
大專生-兼任助理：邱莉婷
大專生-兼任助理：吳韶瑋
大專生-兼任助理：楊紘

中 華 民 國 106 年 10 月 02 日

中 文 摘 要 ： 等待時間變異（WTV）定義為顧客對於兩道菜之間隔等待時間的變異數。本研究首創從典型的台式餐廳中，提出一個等待時間變異最小化之單機排程問題。對於等待時間變異最小化的單機排程問題而言，V字形性質是一個非常重要的最佳解性質。V字形性質是指在處理時間最小的工件之前，以處理時間遞減方式來排序；而在最小的工件之後則以遞增方式來排序，以獲致最小化等待時間變異的效果。但V字形性質在開始的前幾個等待間隔卻會耗費大量時間。為了提供顧客公平且一致的服務，本研究將由V字形性質延伸題出一個新的鋸齒狀A字形啟發式演算法。此演算法將處理時間最大的工件之前，以處理時間遞增方式排序，而之後則以遞減方式排序。為了降低整備時間，此演算法得到的排程會大致呈現A字形且具有鋸齒狀的邊緣。此外，本研究問題必須考慮優先次序的限制。本研究問題的目標函數，對於所有顧客而言，是求取等待時間變異的最小化。本研究也發展出一個禁忌搜尋演算法（tabu search）來進一步改善從鋸齒狀A字形啟發式演算法所得到的解。最後，本研究將進行一連串的實驗來評估所提出的各種演算法的績效。

中文關鍵詞： 排程; 等待時間變異; 優先次序限制; V字形啟發式演算法; 鋸齒狀A字形啟發式演算法； 禁忌搜尋演算法

英 文 摘 要 ： Waiting Time Variance (WTV) is defined as the interval of waiting time between serving two dishes for the customers. This research is the first one to explore and address a single machine WTV scheduling problem originated from the restaurant industry. A very important optimality property stating that the optimal solution of single machine WTV scheduling problem should satisfy V-shaped property. The V-shaped property indicates that the jobs before the smallest job are scheduled in a non-increasing order of their processing times, and the jobs after the smallest job are scheduled in a non-decreasing order of their processing times to achieve the effect of minimizing variance. But, a V-shaped schedule will take long time on the initial several intervals. To provide a fair and consistent service for customers, this research will extend the V-shaped property to propose a new "Serrated A-shaped" heuristic that the jobs before the longest job are scheduled in a non-decreasing order of their processing times, and the jobs after the longest job are scheduled in a non-increasing order of their processing times. For reducing the setup time, the schedule may appear a roughly A-shaped with serrated edges. Furthermore, there are precedence constraints which must be considered in the problem. The objective of the single machine WTV scheduling problem is to minimize WTV for all customers. A tabu search meta-heuristic is developed to further improve the solution obtained from Serrated A-shaped heuristic. Finally, some experiments will be conducted to evaluate the performance of the proposed scheduling methods.

英文關鍵詞： scheduling; waiting time variance; precedence constraint; V-shaped heuristic; Serrated A-shaped heuristics; tabu search

行政院國家科學委員會補助專題研究計畫成果報告

（□期中進度報告/■期末報告）

台式餐廳求解等待時間變異最小化之出菜排程問題

計畫類別：■個別型計畫　□整合型計畫
計畫編號：MOST 105－2410－H－263－003－
執行期間：105 年 8 月 1 日至 106 年 7 月 31 日

執行機構及系所：致理科技大學國際貿易系

計畫主持人：李政雄
共同主持人：
計畫參與人員：

本計畫除繳交成果報告外，另含下列出國報告，共 ___ 份：
□執行國際合作與移地研究心得報告
□出席國際學術會議心得報告

期末報告處理方式：
1. 公開方式：
■非列管計畫亦不具下列情形，立即公開查詢
□涉及專利或其他智慧財產權，□一年□二年後可公開查詢
2.「本研究」是否已有嚴重損及公共利益之發現：■否 □是
3.「本報告」是否建議提供政府單位施政參考 ■否 □是，____（請列舉提供之單位；本會不經審議，依勾選逕予轉送）

中　華　民　國　106 年　9　月　27　日

中文摘要

等待時間變異 (WTV) 定義為顧客對於兩道菜之間隔等待時間的變異數。本研究首創從典型的台式餐廳中，提出一個等待時間變異最小化之單機排程問題。對於等待時間變異最小化的單機排程問題而言，V 字形性質是一個非常重要的最佳解性質。V 字形性質是指在處理時間最小的工件之前，以處理時間遞減方式來排序；而在最小的工件之後則以遞增方式來排序，以獲致最小化等待時間變異的效果。但 V 字形性質在開始的前幾個等待間隔卻會耗費大量時間。為了提供顧客公平且一致的服務，本研究將由 V 字形性質延伸題出一個新的鋸齒狀 A 字形啟發式演算法。此演算法將處理時間最大的工件之前，以處理時間遞增方式排序，而之後則以遞減方式排序。為了降低整備時間，此演算法得到的排程會大致呈現 A 字形且具有鋸齒狀的邊緣。此外，本研究問題必須考慮優先次序的限制。本研究問題的目標函數，對於所有顧客而言，是求取等待時間變異的最小化。本研究也發展出一個禁忌搜尋演算法 (tabu search) 來進一步改善從鋸齒狀 A 字形啟發式演算法所得到的解。最後，本研究將進行一連串的實驗來評估所提出的各種演算法的績效。

關鍵詞：排程; 等待時間變異; 優先次序限制; V 字形啟發式演算法; 鋸齒狀 A 字形啟發式演算法； 禁忌搜尋演算法

Abstract

Waiting Time Variance (WTV) is defined as the interval of waiting time between serving two dishes for the customers. This research is the first one to explore and address a single machine WTV scheduling problem originated from the restaurant industry. A very important optimality property stating that the optimal solution of single machine WTV scheduling problem should satisfy V-shaped property. The V-shaped property indicates that the jobs before the smallest job are scheduled in a non-increasing order of their processing times, and the jobs after the smallest job are scheduled in a non-decreasing order of their processing times to achieve the effect of minimizing variance. But, a V-shaped schedule will take long time on the initial several intervals. To provide a fair and consistent service for customers, this research will extend the V-shaped property to propose a new "Serrated A-shaped" heuristic that the jobs before the longest job are scheduled in a non-decreasing order of their processing times, and the jobs after the longest job are scheduled in a non-increasing order of their processing times. For reducing the setup time, the schedule may appear a roughly A-shaped with serrated edges. Furthermore, there are precedence constraints which must be considered in the problem. The objective of the single machine WTV scheduling problem is to minimize WTV for all customers. A tabu search meta-heuristic is developed to further improve the solution obtained from Serrated A-shaped heuristic. Finally, some experiments will be conducted to evaluate the performance of the proposed scheduling methods.

***Keywords*:** scheduling; waiting time variance; precedence constraint; V-shaped heuristic; Serrated A-shaped heuristics; tabu search

台式餐廳求解等待時間變異最小化之出菜排程問題

Dishing-up scheduling problem for minimizing waiting time variance in a Taiwanese restaurant

1. Introduction

The restaurant industry is highly competitive in Taiwan, especially in restaurants of medium and small sizes. In Taiwanese folk custom, people like to have a family dinner party in a restaurant for celebrating good things, such as birthday, promotion and house moving, etc. Generally, there are about from 20 to 50 kinds of dishes in the menu of the medium or small size restaurant. Customers may choose 10 dishes from the menu because the number 10 symbolizes luck, perfect and reunion. Since there are as many as 10 dishes during the entire dining period, customers may spend about two hours for enjoying the happy time.

In this research, we use a typical Taiwanese restaurant as the case restaurant in which it has 5 tables and 30 different dishes in the menu. These 30 dishes have the respective dish number from No.1 to No.30. There is only one chef in the restaurant. Each dish has its own cooking time and must be processed by the chef in the kitchen. The phrase “dish up” means when a dish is finished in the kitchen, it is immediately sent to its respective table. In the dietetic culture of Taiwan, some kinds of dishes have to dish up prior to the others. For example, meat dishes must be prior to soups, and vegetable must be prior to dessert. The numbers of the dishes are from No.1 to No.15 which must dish up before the dishes with No.16 to No.30. Therefore, precedence constraints have to be considered in the restaurant. Further, since the kitchen equipment is limited to the size and capacity, the chef can process at most two same dishes that are merged at a time. The processing time of the merged dishes is shorter than two times of the processing time of the dish. A dish (including the merged dishes) processed by the chef is called a job, which is the basic unit on the scheduling operation. In the kitchen, if the two adjacent jobs have the different dish number, a setup time of 2 minutes will be incurred. This is because the chef must make a setup adjustment of the kitchen equipment whenever there is a switch to a different job. On the contrary, there is no setup time incurred if the two adjacent jobs have the same dish number.

The key factor of customer satisfaction during the dining period is waiting time, which is defined as the time from a job dished up on the table to the next one. Because customers may compare the waiting time with others, a longer waiting time will lead to unfair feelings for customers. Therefore, consistency of the waiting time for each customer is closely connected to Quality of Service (QoS) in the restaurant industry. The unfair feelings of customers directly decrease the Qos level of the restaurant. Waiting Time Variance (WTV) is defined as the interval of waiting time between serving two dishes for the customers. The most important concern is to provide better QoS in restaurant environment. Thus, to measure the waiting time variance for customers is the main interest of this research. We formulate a minimizing WTV job scheduling problem where we schedule the dishes ordered by customers, to be processed on a single machine (i.e., the chef), in such a way that the variance of their waiting times is minimized. Minimizing

WTV is a well-known scheduling problem, important in providing QoS in many industries. This research is the first one to explore the issue of WTV in restaurant industry.

As the earlier research, e.g., Eilon and Chowdhury (1977) and Xu (2011), an important property stating that all the optimal solution of single machine WTV scheduling problem should be V-shaped property. V-shaped property indicates that the jobs before the smallest job are scheduled in a non-increasing order of their processing times, and the jobs after the smallest job are scheduled in a non-decreasing order of their processing times to achieve the effect of minimizing variance (see Figure 1).

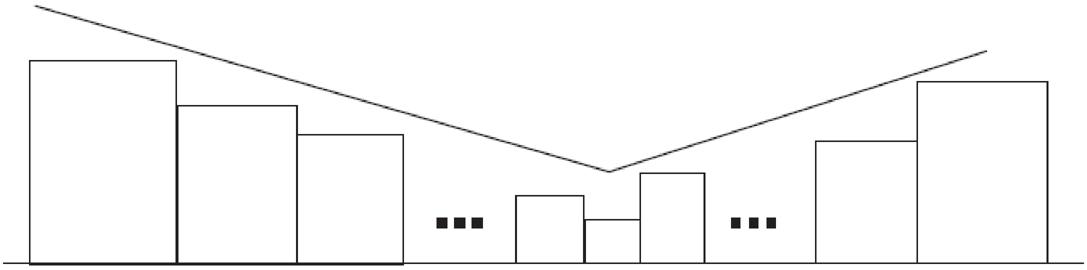


Figure 1. V-shaped property (Xu, 2011)

However, V-shaped arrangement will take longer time on the initial several waiting times. If customers take long waiting time in the beginning of dining period, they will feel that are treated unfairly. The QoS must be reduced for those customers who feel unfair. To keep the level of QoS, this research will extend the V-shaped property to propose a new concept called A-shaped to solve the single machine WTV scheduling problem.

2. Literature review

Regarding to the importance of restaurant management to Quality of Service (QoS), Vidotto et al. (2007) presented a restaurant table management problem which is a complex dynamic combinatorial problem. They defined it as a constraint satisfaction problem and presented a constraint-based solution for enhancing restaurant table management. Hwang and Lambert (2008) examined the impact of major resources on multi-stage waiting times and their interactions on waiting times under the restaurant environment. They showed that each resource influenced waiting for different service stages and that interaction among the multi-resources incurred.

Waiting time variance (WTV) problem is the extension of the completion time variance (CTV) problem. WTV and CTV problems are first introduced by Merten and Muller (1972). They explored the single-machine case, and proved that the minimum of these two variance measures are the same for the same set of jobs, and they showed that the sequence of minimizing CTV is antithetical to the sequence of minimizing WTV. Later, Kubiak (1993) proved that the CTV problem is NP-hard. Ganesan et al. (2006) addressed a static job-shop scheduling problem, subject to the minimum completion time variance (CTV). They developed a lower bound on CTV and proposed a backward scheduling approach for hierarchical minimization of CTV and makespan. Li et al. (2010) proposed a job scheduling problem on multiple identical parallel machines to minimize CTV. They proved several dominant properties about the optimal solution to the problem and developed an efficient heuristic algorithm based on those properties. Srirangacharyulu and Srinivasan (2010) considered the problem of minimizing the CTV on a single machine with deterministic processing times. They proposed a new heuristic and a method based on genetic algorithms to solve the problem. The experiments showed that the proposed methods provide better results compared to existing methods for the single machine case as well as for the multi-machine case.

Minimizing WTV is a well-known scheduling problem, important in providing Quality of Service (QoS) in many industries. There are a lot of works which discuss the solving approaches and industrial applications of the WTV problem. Eilon and Chowdhury (1977) considered the single machine scheduling problem to minimize the WTV. They presented a theorem that the optimal sequence must be V-shaped and proposed a heuristic method for solving the problem where n is large-size. Xu and Ye (2007) dealt with the identical parallel machine scheduling problem to minimize the waiting time variance of the jobs (Pm || WTV) and developed the heuristic algorithms to solve the problems. The testing results of the proposed heuristic algorithms can be applied to the problems with both small and large job sets. Ye et al. (2007) presented that minimizing the WTV on computer networks can lead to stable and predictable network performance. They developed two scheduling heuristics and then they tested and compared the proposed heuristics with four other scheduling methods on both small and large size instances. Li et al. (2007) considered the job scheduling problem of minimizing the weighted waiting time variance (WWTV) which is an extension of WTV minimization problems. They formulated a WWTV problem as an integer programming problem and discovered the strong V-Shape tendency of the optimal job sequences for this problem. They developed two scheduling algorithms for the WWTV problem and showed that the two proposed algorithms significantly outperform existing WWTV algorithms. Xu (2011) considered the single processor scheduling problem which each job has different size and weight. The objective is to minimize the weighted waiting time variance. They showed that the objective function of the problem can be expressed as a function of positional weights and processing times. Amiri et al. (2014) proposed a single machine scheduling problem to minimize a linear combination of total tardiness and WTV in which the idle time is not allowed. Minimizing waiting time variance is an important criterion in establishing quality of service (QoS) in many systems. They developed a genetic algorithm (GA) by applying its general structure that further improves the initial population, utilizing some of heuristic algorithms. The GA is shown to perform well by testing on various instances.

3. Problem formulation

In this section, the following notations are defined and used throughout the paper:

T	Number of tables
n	Number of dishes ordered for each table
K	Number of dishes in menu

$\lvert K \rvert$	Dish set containing all the dish numbers in menu. $\lvert K \rvert = \{1, 2, \cdots, K\}$
$J_{i,k}$	kth dish of menu ordered by ith table in ordering stage, $i = 1, \ldots, T$; $k \in \lvert K \rvert$
$J_{r,k}$	A job with kth dish number scheduled at rth position on single machine in sequencing and processing stage, $r \le Tn$, $k \in \lvert K \rvert$
$J_{i,j}$	jth dish on the ith table in dishing-up stage, $i = 1, \ldots, T$; $j = 1, \ldots, n$
P_k	Processing time of the dish with kth dish number, $k \in \lvert K \rvert$
$P_{r,k}$	Processing time of the job with the kth dish number which is scheduled at rth position of single machine, $r \le Tn$, $k \in \lvert K \rvert$
$C_{r,k}$	Completion time of the job with kth dish number scheduled at the rth position on single machine in sequencing and processing stage, $r \le Tn$, $k \in \lvert K \rvert$
$s_{r,k}$	Setup time of the rth job with kth dish number on single machine, $r \le Tn$, $k \in \lvert K \rvert$
$C_{i,j}$	Completion time of jth dish on the ith table in dishing-up stage, $i = 1, \ldots, T$; $j = 1, \ldots, n$
α	Increasing rate of processing time, $1 \le \alpha < 2$
J^A , J^B	Job sets. All jobs in J^A must be completed before any job of J^B is allowed to start.
$W_{i,j}$	Waiting time of the jth dish on the ith table, $i = 1, \ldots, T$; $j = 1, \ldots, n$

Consider the scheduling problem where there are T tables and n dishes for each table to be processed on a single machine. Here the single machine denotes the single chef of the restaurant. We use three stages to explain the studied problem: ordering stage, sequencing and processing stage, and dishing-up stage (see Table 1). Job may appear in different forms for the three stages. In ordering stage, we define that all the numbers of ordered dishes are the same for each table, i.e., each table has n dishes. There are $T \times n$ dishes to be processed on a single machine. Time zero is defined as when all the customer orders are collected. Job in this stage is denoted as $J_{i,k}$ which is the kth dish of the menu ordered by the ith table.

In sequencing and processing stage, the chef can merge at most two dishes with the same dish number k into a single job which has α times of processing time. Each job has a processing time $P_{r,k}$. If two same dishes are merged as a single job and scheduled at the rth position of the machine, the processing time of the job is $\alpha P_{r,k}$. If the job contains only one dish, $\alpha = 1$; if the job contains two merged dishes, $1 < \alpha < 2$ (In this case, $\alpha = 1.5$). Job in this stage is denoted as $J_{r,k}$ where an ordered dish $J_{i,k}$ is scheduled at the rth position on the single machine. For example, if there are v merged dishes, the number of jobs in this stage is $Tn - v$, where $0 \le v \le \lfloor Tn/2 \rfloor$. Since it is necessary to adjust the kitchen equipment for any job, the first setup time must be incurred before starting the job scheduled on the first position of the machine. If the adjacent two jobs have different dish number (k), a setup time ($s_{r,k}$) of 2 minutes between two adjacent jobs is incurred. Moreover, because certain dishes have to be completed before other dishes are allowed to start processing, precedence constraints are considered in the studied problem. $C_{r,k}$ is defined as the completion time of the job scheduled at the rth position of the machine. Two job sets are defined as

J^A and J^B where all the jobs in J^A must be completed before any job of J^B is allowed to start. For example, J_a is an arbitrary job of J^A and J_b is an arbitrary job of J^B; let C_{J_a} and C_{J_b} denote the completion times of J^A and J^B respectively. The precedence constraints can be obtained that $C_{J_a} < C_{J_b}$. In sequencing and processing stage, a sequence of all jobs must be determined for processing on the single machine.

In dishing-up stage, whenever a job is finished on the machine, it is immediately dished up to its respective table. The time from kitchen to table is omitted. Job in this stage is denoted as $J_{i,j}$ which represents the jth dish on the ith table. $C_{i,j}$ is denoted as the completion time of the jth dish when it is dished up to the ith table. Therefore, there are n waiting times for each table. Note that the first waiting time of each table is computed from time zero to which the first dish is dished up on the table. $W_{i,j}$ is defined as the waiting time of the jth dish on the ith table. The objective of the problem is to minimize total WTV for each table. Following the three-field notation, the problem can be denoted by $1|s_r,\ prec|WTV$, where 1 designates the single machine, s_r and $prec$ represent the setup time and precedence constraints respectively, and WTV denotes the waiting time variance. Since WTV scheduling problem is NP-hard which has been already proven in (Kubiak, 1993), we need to develop a simple and effective method for solving the problem.

The studied scheduling problem can be formulated as follows: In sequencing and processing stage, each job $J_{r,k}$ may contain only one dish or at most two dishes with same number k. Thus, the completion time of $J_{r,k}$ is denoted as $C_{r,k}$ and defined as

$$C_{r,k} = C_{r-1,k} + s_{r,k} + \alpha P_{r,k}$$

where the setup time $s_{r,k}$ is incurred if the dish numbers k is different between $J_{r,k}$ and $J_{r-1,k}$. Otherwise, $s_{r,k} = 0$. The processing time of $J_{r,k}$ is denoted as $P_{r,k}$. If $J_{r,k}$ contains only one dish, $\alpha = 1$; if two merged dishes, $1 < \alpha < 2$. Moreover, the precedence constraints must be considered in this stage: $C_{J^B} > C_{J^A}$, where C_{J^A} and C_{J^B} denote the completion time of a job belonging to the job set J^A or J^B, respectively. The precedence constraints explain that the processing time of any job belonging to J^B must be greater than that belonging to J^A.

Whenever a job $J_{r,k}$ is completed on the single machine, it is immediately dished up to the respective table. Thus, $C_{r,k} = C_{i,j}$ if $J_{r,k}$ contains the jth dish of the ith table. But, if $J_{r,k}$ contains two dishes (e.g., $J_{1,k}$ and $J_{2,k}$) which have the same dish number k and the different table i, $C_{r,k}$ represents the completion time of two dishes with the same dish number k on table 1 and 2. Therefore, the waiting time of the jth dish on the ith table can be defined as

$$W_{i,j} = C_{i,j} - C_{i,j-1}, \text{ and } C_{i,0} = 0$$

For each table, the average waiting time of the jth dish can be computed as

$$\overline{W}_{i,j} = \frac{1}{T}\sum_{i=1}^{T} W_{i,j} = \frac{1}{T}\sum_{i=1}^{T}(C_{i,j} - C_{i,j-1}), \text{ for } j = 1, \dots, n$$

Then, the waiting time variance of the jth dish (WTV) can be calculated as

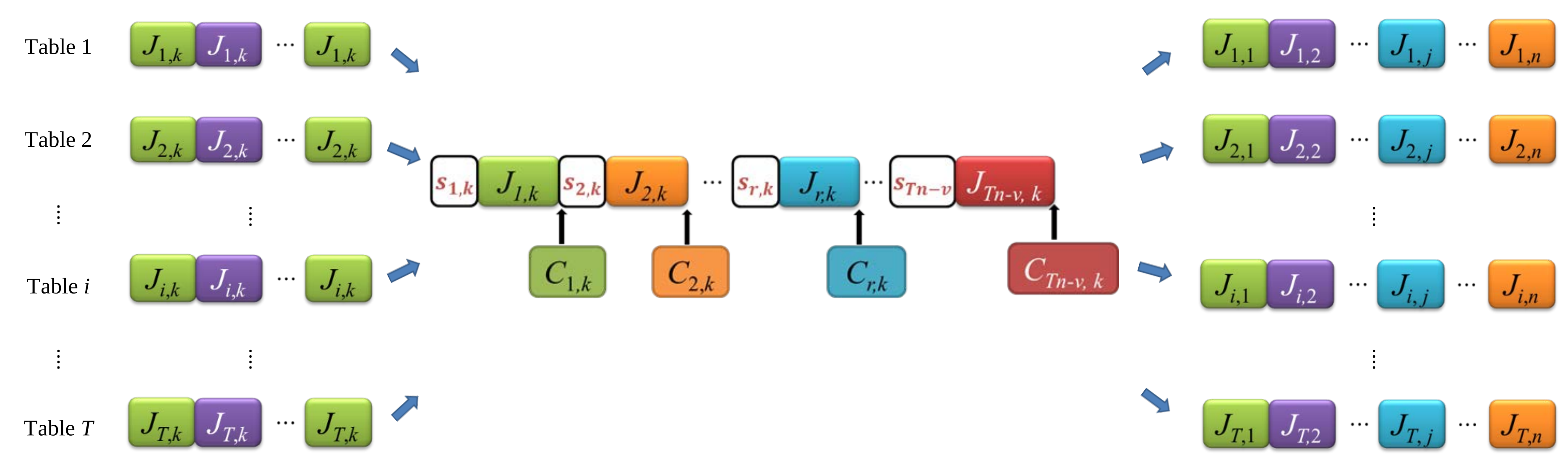
$$WTV = \frac{1}{T-1}\sum_{i=1}^{T}\left(W_{i,j} - \overline{W}_{i,j}\right)^2, \text{ for } j = 1, \dots, n$$

Finally, the objective function (Z) of the studied problem is to minimize the waiting time variance for all tables. The objective function (Z) can be defined as

$$\text{Minimize } Z = \frac{1}{n}\sum_{j=1}^{n}\left[\frac{1}{T-1}\sum_{i=1}^{T}(W_{i,j} - \overline{W}_{i,j})^2\right]$$

Table 1. Three stages of the case restaurant

Ordering stage	Sequencing & processing stage	Dishing-up stage
• Job: $J_{i,k}$. • There are n different dishes ordered by each table. • There are $T \times n$ dishes to be collected.	• Job: $J_{r,k}$, and setup time: $s_{r,k} \in \{0, 2\}$. • At most two dishes ($J_{i,k}$) with the same k can be merged into a single job with 1.5 times of processing time (i.e., $\alpha P_{r,k}, \alpha = 1.5$). • Suppose that there are v merged jobs, $0 \le v \le \left\lfloor \frac{Tn}{2} \right\rfloor$. • Chef must determine a sequence to process the $Tn - v$ jobs. • Each job can obtain its own completion time $C_{r,k}$.	• Job: $J_{i,j}$. • According to the sequence determined by the chef, each finished dish is sent to its respective table, i.e., $J_{i,j}$ is the jth dish on table i.



4. Current scheduling method in case restaurant

The scheduling problem in the case restaurant involves processing jobs by a single chef to determine a job sequence for processing all the dishes. The objective of the problem is to minimize the waiting time variance for all tables. The chef needs to consider some characteristics in the case restaurant:

- Precedence constraints: Certain jobs have to be completed before other jobs are allowed to start processing.
- Setup time: If the adjacent two jobs have different dish number, a setup time between two adjacent jobs is incurred.
- Two dishes may be merged into one: The chef can merge at most two same dishes into a single job with α times of the processing time.

Furthermore, two job sets are defined as J^A and J^B where all the jobs in J^A must be completed before any job of J^B is allowed to start. The chef in the case restaurant also realizes the concept of V-shaped property which can provide a good outcome in minimizing the waiting time variance. According to these considerations and the chef's previous work experience, the current scheduling method in case restaurant is developed as the following steps:

Step 1 Collect the customer orders from each table. Divide all dishes into two sets: J^A and J^B where the dishes in J^A must be completed before any dish of J^B is allowed to start processing.

Step 2 For the dishes with the same number of k in J^A and J^Brespectively, merge the same dishes, two by two, into a single job until no more jobs can be merged. Each of the unmerged dishes is regarded as a single job. Calculate the processing time P_k for all jobs.

Step 3 According to the concept of V-shaped property, for jobs in J^A, sequence them in no-increasing order of the processing time P_k. And then, for jobs in J^B, sequence them in no-decreasing order of the processing time P_k. Based on the schedule, whenever a job is completed by the chef, it is immediately dished up to the respective table.

The current scheduling method of the case restaurant is briefly explained as the following. In the first step, the chef has to divide all the dishes ordered by customers into two sets due to the precedence constraints. In Step 2, the chef tries to do the best for merging the same dishes in order to reduce the processing times. In Step 3, the chef uses the concept of V-shaped property to obtain a schedule in the kitchen.

（三）研究方法、進行步驟及執行進度

5. Proposed heuristic

Although V-shaped property is an important optimality property for the single machine WTV scheduling problem, V-shaped arrangement will take longer time on the initial several waiting times. In the beginning of dining period, customers very care about a fair and consistent service. If customers take long waiting time in the beginning of dining period, they will feel that are treated unfairly. Thus, the QoS must be reduced for those customers who feel unfair. This research will extend the V-shaped property to propose a new concept called A-shaped. The A-shaped concept can reduce the initial several waiting times that the jobs before the longest job are scheduled in non-decreasing order of their processing times, and the jobs after the longest job are scheduled in non-increasing order of their processing times. Subject to the precedence constraints and setup times, an effective heuristic, called Serrated A-shaped heuristic, is developed to solve the studied scheduling problem and described as follows:

Step 1 Collect all $J_{i,k}$ and merge the same $J_{i,k}$, two by two, into a single job $J_{r,k}$ until no more $J_{i,k}$ can be merged. Calculate P_k for all $J_{r,k}$.

Step 2 Divide all $J_{r,k}$ into J^A and J^B. All $J_{r,k}$ in J^A must be completed before any $J_{r,k}$ of J^B is allowed to start processing.

Step 3 Set $r = 1$. Choose a $J_{r,k}$ with the smallest P_k from J^A and assign it at the first position of the machine. Remove $J_{1,k}$ from J^A.

Step 4 Set $r = r + 1$. If $J^A \in \emptyset$, go to Step 5.

Step 4.1 If there is no job with the same k as $J_{r,k}$ in J^A, assign a job with the smallest P_k from J^A to the machine and remove it from J^A. Go to Step 4.

Step 4.2 If there is still only one job with the same k as $J_{r,k}$, assign it at the position immediately adjacent to $J_{r,k}$. Remove the assigned job from J^A. Go to Step 4.

Step 4.3 If there are still more than one jobs (e.g., m jobs) with the same k as $J_{r,k}$, assign them, in non-decreasing order of their processing times, at the next positions immediately adjacent to $J_{r,k}$. Remove the assigned jobs from J^A. Set $r = r + m$ and go to Step 4.

Step 5 Set $r = r + 1$. Choose a job with the longest P_k from J^B and assign it at the next position of the machine. Remove it from J^B. If $J^B \in \emptyset$, go to Step 6.

Step 5.1 If there is no job with the same k as $J_{r,k}$ in J^B, assign a job with the longest P_k from J^B to the machine and remove it from J^B. Go to Step 5.

Step 5.2 If there is still only one job with the same k as $J_{r,k}$, assign it at the position immediately adjacent to $J_{r,k}$. Remove the assigned job from J^B. Go to Step 5.

Step 5.3 If there are still more than one jobs (e.g., m jobs) with the same k as $J_{r,k}$, assign them, in non-increasing order of their processing times, at the next positions immediately adjacent to $J_{r,k}$. Remove the assigned jobs from J^B. Set $r = r + m$ and go to Step 5.

Step 6 Stop the procedure. Each completed job $J_{i,j}$ is immediately dished up to the respective ith table. Compute the waiting time variance of the jth dish (WTV) for all tables to be the final solution Z.

Since the precedence constraints are existed, i.e., certain dishes have to be completed before other dishes are allowed to start processing, the A-shaped property is naturally suitable to the studied problem. Now the proposed Serrated A-shaped heuristic is briefly explained as follows. In Step 1, two same dishes are merged as possible so that the processing times can be reduced. Step 2 states the precedence constraints. Step 3 determines how to assign the first job to the first position on the machine. Step 4 and Step 5 describe how to assign the jobs in J^A and J^B to the machine until the job sets J^A and J^B are empty. Also, these two steps try to reduce the setup times as much as possible. In Step 6, the WTV of each dish of each table can be computed as the objective function Z. After implementing Step 4 and Step 5, it can be observed that the final job schedule on the machine may appear a roughly "A" in shape with serrated edges. The process of setup time reduction may result in the serrated edges of A-shaped (see Figure 2).

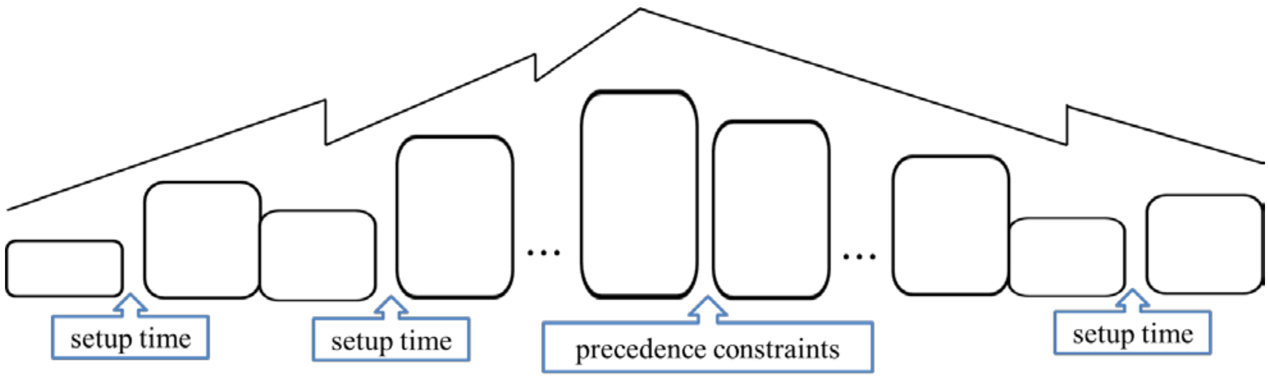


Figure 2. Property of serrated A-shaped

The current scheduling method considers the concept of V-shaped property and the precedence constraints, but the consideration of setup time is ignored. The proposed Serrated A-shaped heuristic not only considers reducing the initial several waiting times to keep the level of QoS, but also tries to reduce the setup times as much as possible. Consequently, by comparison with the V-shaped property, the proposed Serrated A-shaped heuristic must be more applicable for the studied scheduling problem.

6. Developed tabu search

To further improve the performance of the proposed Serrated A-shaped heuristic, a tabu search (TS) is developed. TS, firstly proposed by Glover (1989), is one of the most popular meta-heuristics for a large number of combinational problems. It can be applied in various scheduling problems. The idea of TS is to chase a local optimum by allowing a non-improving move. In general, there are six elements in TS: initial solution, stopping criterion, neighborhood structure, aspiration criteria, tabu size and long-term search strategy. TS starts from an initial solution until stopping criterion is fulfilled. The neighborhood structure is defined as a space of all feasible solutions. The aspiration criterion is used to revoke a tabu list when there is an attractive tabu move. Tabu size is utilized to escape from local optimum and cycling search. The long-term search strategy is employed as a searching strategy in the solution space. Here we refer the TS of Lee et al. (2012) which is developed for solving the single machine scheduling problem. The structure of the developed TS is described as follows:

Step 1 Generate an initial sequence from the Serrated A-shaped heuristic.

Step 2 Obtain neighborhood sequences by the current sequence by exchanging the position of two jobs in all possible ways under the precedence constraints. Compute WTV for each obtained sequence to be the solution.

Step 2.1 Consider all the neighborhood sequences. If the tabu move exists in the tabu list, then the sequence is not considered as a candidate sequence unless it is better than the incumbent best solution. Otherwise, the neighborhood sequence is considered as a candidate sequence.

Step 2.2 Let the sequence with the minimum solution as the current sequence from all the candidate sequences.

Step 2.3 Add the tabu move of the current sequence in the tabu list.

Step 2.4 If the solution of the current sequence is better than the incumbent best sequence, then update the incumbent best sequence.

Step 2.5 If the number of iterations exceeds the pre-set maximum number, then continue. Otherwise, go to Step 2.

Step 3 Select the best solution from all the current sequences.

In the developed TS, a sequence is obtained in sequencing and processing stage, then computing the objective value (WTV) in dishing-up stage. The sequence obtained from Serrated A-shaped heuristic is applied as the initial solution of the TS, the neighborhood is the set of all the sequences obtained by the current sequence by exchanging the position of two jobs in all possible ways under the precedence constraints, the stopping criterion is to use a fixed number of 10,000 iterations or the best solution is not improved within 1,000 iterations, the aspiration criterion is to accept a solution better than the incumbent best solution even if the solution is generated by a tabu move, the tabu size is $3Tn$, and the pair interchange is used to improve the solution.

7. Computational experiments

The parameter combinations of the studied scheduling problem are set as $T = \{5, 10, 20\}$,

$n = \{10, 20, 50\}$, $K = 100$, $P_k = (5, 50)$, $\alpha = 1.5$. And then, a series of experiments will be conducted to investigate the aspects of the studied problems and briefly stated as follows:

Experiment 1: Compare the effectiveness of Serrated A-shaped heuristic with the current scheduling method (CM) based on the concept of V-shaped property.

To investigate the effectiveness and efficiency of the proposed Serrated A-shaped heuristic, it is compared with CM. In this experiment, the test problem sizes are generated with number of jobs $n = \{20, 20, 50\}$. The percentage improvements (PI) are calculated as below:

$$\mathrm{PI} = \frac{(\text{solution of CM}) - (\text{solution of A} - \text{shaped})}{\text{solution of CM}} \times 100\%$$

The computational times (in seconds) and percentage improvements are summarized in Table 2. It can be observed from Table 2 that the Serrated A-shaped heuristic outperforms the current method since all the values of PI are positive. Regarding to computational time, although they increase significantly as the number of jobs increase, both Serrated A-shaped and CM take a little time due to their simplicity. The percentage improvements increase significantly from 4.84% to 21.67% as n increases for all the test instances. The average improvement is 12.01% for all the test instances. We notice that the test instances with $T = 5$ can obtain a larger PI than those instances with $T = 10$ and 20. It is because the instances with more tables could be more difficult to improve. To sum up, the Serrated A-shaped heuristic is more effective than CM.

Table 2. Comparative results of Serrated A-shaped heuristic and the current scheduling method

		A-shaped		CM
n	*T*	PI	Time	Time
10	5	6.48	0.00	0.00
	10	4.79	0.00	0.00
	20	3.25	0.01	0.00
	Average	4.84	0.00	0.00
20	5	9.13	0.00	0.00
	10	13.82	0.01	0.00
	20	5.65	0.01	0.00
	Average	9.53	0.00	0.00
50	5	26.28	0.00	0.00
	10	22.17	0.00	0.00
	20	16.55	0.01	0.01
	Average	21.67	0.01	0.00
Agg. average		12.01	0.00	0.00

Experiment 2: Compare the Serrated A-shaped heuristic with an existing DBS heuristic for evaluating both efficiency and effectiveness.

To investigate the effectiveness and efficiency of the proposed Serrated A-shaped heuristic, it is compared with an existing dynamic balanced spiral (DBS) heuristic (Xu and Ye, 2007). In this experiment, the test problem sizes are also generated with number of jobs $n = \{20, 20, 50\}$. The percentage improvements (PI) are computed as below:

$$\mathrm{PI} = \frac{(\text{solution of DBS}) - (\text{solution of A} - \text{shaped})}{\text{solution of DBS}} \times 100\%$$

The computational times (in seconds) and percentage improvements are summarized in Table 3. It can be observed from Table 3 that the Serrated A-shaped heuristic outperforms the DBS since all the values of PI are almost positive. Regarding to computational time, although they increase significantly as the number of jobs increase, both Serrated A-shaped and DBS take a little time. The percentage improvements decrease significantly from 3.87% to 1.41% as *n* increases for all the test instances. The average improvement is 2.66% for all the test instances. It can be concluded that the Serrated A-shaped heuristic is more effective than DBS.

Table 3. Comparative results of Serrated A-shaped heuristic and the DBS algorithm

		A-shaped		DBS
n	*T*	PI	Time	Time
10	5	3.22	0.00	0.01
	10	4.68	0.00	0.01
	20	3.71	0.01	0.01
	Average	3.87	0.00	0.01
20	5	2.93	0.00	0.01
	10	2.64	0.01	0.01
	20	2.51	0.01	0.01
	Average	2.69	0.01	0.01
50	5	-0.42	0.00	0.01
	10	2.09	0.01	0.01
	20	2.57	0.01	0.02
	Average	1.41	0.01	0.01
	Agg. average	2.66	0.01	0.01

Experiment 3: Compare the Serrated A-shaped heuristic with the developed tabu search to evaluate the improvement of the solution quality.

To investigate the performance of the proposed tabu search (TS), it is compared with the Serrated A-shaped heuristic in order to evaluate the improvements of the initial solutions. In this experiment, the test problem sizes are generated with number of jobs $n = \{20, 40, 80, 100\}$. For each test instance, the proposed TS is run 10 times to obtain an average solution value. The percentage improvements (PI) are calculated as follows:

$$\mathrm{PI} = \frac{(\text{solution of A} - \text{shaped}) - (\text{solution of TS})}{\text{solution of A} - \text{shaped}} \times 100\%$$

The solutions of the Serrated A-shaped heuristic are used to be the initial solutions. The computational times (in seconds) and percentage improvements are summarized in Table 4. If the value of PI is greater than zero, it implies that the TS provides a further improvement for the initial solution obtained by the Serrated A-shaped heuristic. As observed from Table 4, the values of PI decrease significantly as the number of jobs increases for all the test instances. Regarding to the computational time, the TS's is stably and slightly increased as the number of jobs increases. The computational time of the Serrated A-shaped heuristic also almost takes no time. To sum up, the proposed TS can make significant improvement when starting with the Serrated A-shaped heuristic.

Table 4. Improvements of the initial solutions by implementing TS

		TS		A-shaped
n	*T*	PI	Time	Time
10	5	4.52	0.35	0.00
	10	5.17	0.29	0.00
	20	4.23	0.31	0.01
	Average	4.64	0.32	0.00
20	5	3.44	0.81	0.00
	10	2.93	0.66	0.01
	20	2.24	1.07	0.01
	Average	2.87	0.85	0.00
50	5	2.75	1.72	0.00
	10	2.08	1.29	0.01
	20	1.16	2.31	0.01
	Average	2.00	1.77	0.01
Agg. average		3.17	0.98	0.01

8. Conclusions and future research

In this research we have addressed a single machine Waiting Time Variance (WTV) scheduling problem originated from the restaurant industry. An optimality property stating that the optimal solution of single machine WTV scheduling problem should satisfy V-shaped property. The V-shaped property indicates that the jobs before the smallest job are scheduled in a non-increasing order of their processing times, and the jobs after the smallest job are scheduled in a non-decreasing order of their processing times to achieve the effect of minimizing variance. But, a V-shaped schedule will take long time on the initial several intervals. To provide a fair and consistent service for customers, this research will extend the V-shaped property to propose a new "Serrated A-shaped" heuristic that the jobs before the longest job are scheduled in a non-decreasing order of their processing times, and the jobs after the longest job are scheduled in a non-increasing order of their processing times. For reducing the setup time, the schedule may appear a roughly A-shaped with serrated edges. Furthermore, there are precedence constraints which must be considered in the problem. The objective of the single machine WTV scheduling problem is to minimize WTV for all customers. A tabu search is developed to further improve the solution obtained from Serrated A-shaped heuristic.

Further research may be conducted to consider some other factors in the practical restaurant environment. It is also worthwhile to develop a more efficient heuristic for the scheduling problem in the case restaurant.

References

Amiri, M., Olfat, L., & Ghorabaee, M. K. (2014). Simultaneous minimization of total tardiness and waiting time variance on a single machine by genetic algorithms. *The International Journal of Advanced Manufacturing Technology*, 72(1-4), 439-446.

Eilon, S., & Chowdhury, I. G. (1977). Minimising waiting time variance in the single machine problem. *Management Science*, 23(6), 567–575.

Ganesan, V. K., Sivakumar, A. I., & Srinivasan, G. (2006). Hierarchical minimization of completion time variance and makespan in jobshops. *Computers & Operations Research*, 33(5), 1345–1367.

Glover, F. (1989). Tabu search-part I. *ORSA Journal on computing*, 1(3), 190–206.

Hwang, J., & Lambert, C. U. (2008). The interaction of major resources and their influence on waiting times in a multi-stage restaurant. *International Journal of Hospitality Management*, 27(4), 541–551.

Kubiak, W. (1993). Completion time variance minimization on a single machine is difficult. *Operations Research Letters*, 14(1), 49–59.

Lee, C. H., Liao, C. J., and Chao, C. W. (2012). Scheduling with multi-attribute setup times. *Computers & Industrial Engineering*, 63(2), 494–502.

Li, X., Chen, Y., & Sun, Y. (2010). Minimizing job completion time variance for service stability on identical parallel machines. *Computers & Industrial Engineering*, 58(4), 729–738.

Li, X., Ye, N., Liu, T., & Sun, Y. (2007). Job scheduling to minimize the weighted waiting time variance of jobs. *Computers & Industrial Engineering*, 52(1), 41–56.

Merten, A. G., & Muller, M. E. (1972). Variance minimization in single machine sequencing problems. *Management Science*, 18(9), 518–528.

Srirangacharyulu, B., & Srinivasan, G. (2010). Completion time variance minimization in single machine and multi-machine systems. *Computers & Operations Research*, 37(1), 62-71.

Vidotto, A., Brown, K. N., & Beck, J. C. (2007). Managing restaurant tables using constraints. *Knowledge-Based Systems*, 20(2), 160–169.

Xu, X. (2011). Minimizing weighted waiting time variance on a single processor. *Computers & Industrial Engineering*, 61(4), 1233-1239.

Xu, X., & Ye, N. (2007). Minimization of job waiting time variance on identical parallel machines. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 37(5), 917–927.

Ye, N., Li, X., Farley, T., & Xu, X. (2007). Job scheduling methods for reducing waiting time variance. *Computers & Operations Research*, 34(10), 3069–3083.

105年度專題研究計畫成果彙整表

<table>
<tr><td colspan="4">**計畫主持人：**李政雄</td><td colspan="3">**計畫編號：**105-2410-H-263-003-</td></tr>
<tr><td colspan="7">**計畫名稱：**台式餐廳求解等待時間變異最小化之出菜排程問題</td></tr>
<tr><td colspan="5">成果項目</td><td>量化</td><td>單位</td><td>質化
（說明：各成果項目請附佐證資料或細項說明，如期刊名稱、年份、卷期、起訖頁數、證號...等）</td></tr>
<tr><td rowspan="17">國內</td><td rowspan="6">學術性論文</td><td colspan="3">期刊論文</td><td>0</td><td rowspan="2">篇</td><td></td></tr>
<tr><td colspan="3">研討會論文</td><td>0</td><td></td></tr>
<tr><td colspan="3">專書</td><td>0</td><td>本</td><td></td></tr>
<tr><td colspan="3">專書論文</td><td>0</td><td>章</td><td></td></tr>
<tr><td colspan="3">技術報告</td><td>0</td><td>篇</td><td></td></tr>
<tr><td colspan="3">其他</td><td>0</td><td>篇</td><td></td></tr>
<tr><td rowspan="9">智慧財產權及成果</td><td rowspan="3">專利權</td><td rowspan="2">發明專利</td><td>申請中</td><td>0</td><td rowspan="9">件</td><td></td></tr>
<tr><td>已獲得</td><td>0</td><td></td></tr>
<tr><td colspan="2">新型/設計專利</td><td>0</td><td></td></tr>
<tr><td colspan="3">商標權</td><td>0</td><td></td></tr>
<tr><td colspan="3">營業秘密</td><td>0</td><td></td></tr>
<tr><td colspan="3">積體電路電路布局權</td><td>0</td><td></td></tr>
<tr><td colspan="3">著作權</td><td>0</td><td></td></tr>
<tr><td colspan="3">品種權</td><td>0</td><td></td></tr>
<tr><td colspan="3">其他</td><td>0</td><td></td></tr>
<tr><td rowspan="2">技術移轉</td><td colspan="3">件數</td><td>0</td><td>件</td><td></td></tr>
<tr><td colspan="3">收入</td><td>0</td><td>千元</td><td></td></tr>
<tr><td rowspan="15">國外</td><td rowspan="6">學術性論文</td><td colspan="3">期刊論文</td><td>0</td><td rowspan="2">篇</td><td></td></tr>
<tr><td colspan="3">研討會論文</td><td>0</td><td></td></tr>
<tr><td colspan="3">專書</td><td>0</td><td>本</td><td></td></tr>
<tr><td colspan="3">專書論文</td><td>0</td><td>章</td><td></td></tr>
<tr><td colspan="3">技術報告</td><td>0</td><td>篇</td><td></td></tr>
<tr><td colspan="3">其他</td><td>0</td><td>篇</td><td></td></tr>
<tr><td rowspan="9">智慧財產權及成果</td><td rowspan="3">專利權</td><td rowspan="2">發明專利</td><td>申請中</td><td>0</td><td rowspan="9">件</td><td></td></tr>
<tr><td>已獲得</td><td>0</td><td></td></tr>
<tr><td colspan="2">新型/設計專利</td><td>0</td><td></td></tr>
<tr><td colspan="3">商標權</td><td>0</td><td></td></tr>
<tr><td colspan="3">營業秘密</td><td>0</td><td></td></tr>
<tr><td colspan="3">積體電路電路布局權</td><td>0</td><td></td></tr>
<tr><td colspan="3">著作權</td><td>0</td><td></td></tr>
<tr><td colspan="3">品種權</td><td>0</td><td></td></tr>
<tr><td colspan="3">其他</td><td>0</td><td></td></tr>
</table>

	技術移轉	件數	0	件	
		收入	0	千元	
參與計畫人力	本國籍	大專生	4	人次	參與本計畫的四位大專學生，都獲得基礎的學術與行政訓練，對於他們日後無論升學或就業都能有正面的助益。
		碩士生	0		
		博士生	0		
		博士後研究員	0		
		專任助理	0		
	非本國籍	大專生	0		
		碩士生	0		
		博士生	0		
		博士後研究員	0		
		專任助理	0		
其他成果（無法以量化表達之成果如辦理學術活動、獲得獎項、重要國際合作、研究成果國際影響力及其他協助產業技術發展之具體效益事項等，請以文字敘述填列。）					

科技部補助專題研究計畫成果自評表

請就研究內容與原計畫相符程度、達成預期目標情況、研究成果之學術或應用價值（簡要敘述成果所代表之意義、價值、影響或進一步發展之可能性）、是否適合在學術期刊發表或申請專利、主要發現（簡要敘述成果是否具有政策應用參考價值及具影響公共利益之重大發現）或其他有關價值等，作一綜合評估。

1. 請就研究內容與原計畫相符程度、達成預期目標情況作一綜合評估

■達成目標

□未達成目標（請說明，以100字為限）

　　□實驗失敗

　　□因故實驗中斷

　　□其他原因

說明：

2. 研究成果在學術期刊發表或申請專利等情形（請於其他欄註明專利及技轉之證號、合約、申請及洽談等詳細資訊）

論文：□已發表　□未發表之文稿　■撰寫中　□無

專利：□已獲得　□申請中　■無

技轉：□已技轉　□洽談中　■無

其他：（以200字為限）

已著手撰寫論文投稿至相關國際期刊。

3. 請依學術成就、技術創新、社會影響等方面，評估研究成果之學術或應用價值（簡要敘述成果所代表之意義、價值、影響或進一步發展之可能性，以500字為限）

主要應用價值在於應能提升服務業的服務水準、減少顧客整體的等待時間變異。

4. 主要發現

本研究具有政策應用參考價值：■否　□是，建議提供機關

（勾選「是」者，請列舉建議可提供施政參考之業務主管機關）

本研究具影響公共利益之重大發現：□否　□是

說明：（以150字為限）

無。