行政院國家科學委員會專題研究計畫　期中進度報告

總計劃(2/3)

計畫類別：整合型計畫
計畫編號：NSC93-2213-E-002-041-
執行期間：93年08月01日至94年07月31日
執行單位：國立臺灣大學工業工程學研究所

計畫主持人：張時中
共同主持人：廖大穎，陳正剛，林佑昇，周雍強
計畫參與人員：范治民、藍俊宏、高慶騰、何思賢

報告類型：精簡報告

處理方式：本計畫可公開查詢

中　華　民　國 94年6月2日

12吋晶圓製造服務佳化工具與標竿環境之研發(2/3)
Design and Development of Benchmark Environment and Optimization Tools for 300mm Foundry Manufacturing Service(2/3)

計畫編號：93-2213-E-002-041
主持人：張時中教授
共同主持人：周雍強教授、陳正剛副教授、廖大穎助理教授、
林佑昇副教授
計畫參與人員：范治民博士、藍俊宏、高慶騰、何思賢
執行期限：93 年8 月1 日至94 年7 月31 日
執行單位：國立臺灣大學工業工程學研究所

中文摘要

本計劃在第二年度的晶圓製造服務最佳化工具標竿環境之研發上，總計完成了下列四項成果以支援閉迴路的系統控管機制：

A1. 針對三種特定的系統角色（子計畫主持人、網站管理員、系統使用者），設計三種層級的使用流程（協同合作、模擬分析、一般操作）；
A2. 開發一個具備擴充能力的標竿系統應用框架；
A3. 開發一個彈性整合各式最佳化工具的系統架構；
A4. 開發一個標竿環境網站，其中包含各子計劃人員的成果資訊管理、客製化網頁編輯器、網站瀏覽結構管理、系統瀏覽介面、以及通用的程式整合工具。

本系統允許個別子計畫主持人藉由網路登入、編輯並管理其最新之研究結果，以供使用者參考應用。目前各子計畫開發了下列應用工具，提供線上使用與基準比較：

1) 半導體供應鏈之協同排程工具；
2) 針對動態生產排程之排序佳化模擬演算系統；
3) 自動化物料搬運系統中，以差異化物料搬運服務爲目標之差異化先佔派工策略與模擬器；
4) 以最小信賴區間爲基準之循序實驗設計，找尋多步驟製程最佳化設定之工具。

Abstract

The second year designs and developments of an environment for benchmarking optimization tools of foundry manufacturing services have accomplished four items to support the close-loop-control mechanism:

A1. three levels of processes (collaborative, analytic, and operational) for three specific roles (subproject owner, website manager, and system visitor),
A2. a framework for scalability
A3. the architecture for flexible integration of optimization tools,
A4. web-based implementations of project repository management, customized page editor, website structure management, website explorer and generic tool adaptor.

This system allows individual subproject owners to incorporate and manage their latest research results of optimization for benchmarking. Specific tools constructed by individual subprojects for benchmarking include

1) a collaborative scheduling tool for semiconductor supply chain,
2) an ordinal optimization-based policy iteration algorithm for dynamic composition of production schedules,
3) a differentiated pre-emptive AMHS dispatching policy and simulator for differentiated material handling services, and
4) a confidence interval minimization-based sequential experiment design procedure for the optimization of multi-step processes.

一、研究動機與目的

Motivated by the needs for advanced optimization methods and the uniqueness and strength of Taiwan's foundry manufacturing, in this three-year project, we have been designing and developing not only a set of optimization tools but also a benchmark environment for 300mm foundry manufacturing service provisioning. Our objective has three folds:

1. design and development of **advanced optimization tools** to facilitate an efficient, quality and reliable service,
2. establishment of **model and data bases for optimization**, which expose industrial people to the underlying fab characteristics and provide the academia with realistic data for research, and
3. construction of **an easily accessible benchmark environment** for manufacturing service optimization.

To achieve the objective, there are one main project and four subprojects on optimization of foundry services:

Subproject 1: Research on Enabling Technologies for Collaborative Planning and Scheduling in Semiconductor Manufacturing;

Subproject 2: Research on Simulation-based Ordinal Optimization Methods with Applications to Production Scheduling of 300mm Foundry Fabs;

Subproject 3: Shop Floor Control of Hot Lots in 300mm Wafer Manufacturing;
Subproject 4: Evolutionary Process Optimization Methods for Multi-stage Manufacturing Systems with Applications to Semiconductor Yield Rump-up.
The main project aims at establishing a benchmark environment and integrating the data sets, models and optimization tools of individual subprojects into it.

In the first year, we constructed a simple skeleton of the benchmark system, which includes adopting object-oriented technology for system analysis, design and software implementation. The configuration of the benchmark environment included a web server, one database server, and one optimization tool server. Within the database server, PostgreSQL was adopted as the database manipulation software. Our web server adopted Apache as the web server and will use PHP as the scripting language to connect the web page stored in the web server to the database server. PHP also added interaction functionality to our web page to users from outside. The web site was built with HTML language with PHP embedded to enhance its functionality. Other web page techniques, such as java script, CSS, may also be added in the process of website building.

二、研究進度與成果

I. Introduction

The major achievements in the second year of this main project are the constructions of **the architecture for flexible optimization-tools integration** and **a framework for scalable benchmark-environment development**. In addition, on top of the two achievements, we further design and develop **a benchmark system to support the processes in operational and analytical levels [1]**.

During the past two years, four subprojects aiming to the optimizations for 300mm foundry manufacturing service provisions are in proceeding. It raises two challenges to concurrently develop the benchmark system while the optimization methodologies and tools in each subproject are under development.

C1. **How to integrate heterogeneous optimization tools into the benchmark system?** In specific, these optimization tools maybe individually developed in each subproject or contributed by other organizations.

C2. **What is the most efficient way to continuously enrich the benchmark system?** In specific, the benchmark system should easily integrate all the up-to-date research results from each subproject and feedbacks from system visitors.

To cope with C1, the architecture for integrating heterogeneous tools is designed and implemented so that the benchmark system manager can flexibly access and manage these tools. The key for heterogeneous tools integration is the **design of a generic tool-adaptor**, based on which three extension adaptors are developed to support the integration of execution files, dynamic linking libraries, and web service.

To face with C2, a framework for developing a scalable benchmark system is designed and constructed so that each subproject owner can individually design and manage their latest research results on the benchmark system. As demonstrated in Fig 1, it is designed to **support the continuous benchmark and improvement in optimization tools** for 300mm foundry manufacturing services. In addition, the development of framework aims to supporting the close-loop-control mechanism providing three levels of processes (**collaborative, analytic, and operational**) for three specific roles (**subproject owner, website manager, and system visitor**).

Overall, specific achievements in the second year of this main project are as follows:

A1 Design three levels of processes (collaborative, analytic, and operational) for three specific roles (subproject owner, website manager, and system visitor) to support the close-loop-control mechanism for the management and continuous improvement of benchmark system.

A2 Design and develop a framework for scalable benchmark environment development.

A3 Design and develop the architecture for flexible optimization tools integration.

A4 Design and develop a benchmark system on top of A2 and A3 to integrate the processes in operational and analytical levels.

- A4.1 A web-based project repository management environment for subproject owners to manage their files of research results.
- A4.2 A web-based customized page editor for subproject owners to directly edit their research results in html format.
- A4.3 A website structure management mechanism for the binding of the website structure nodes with the files in project repository.
- A4.4 A website explorer to guide user exploring the website.
- A4.5 A generic tool adaptor to support the integration of execution files, dynamic linking libraries, and web service.

II. Design and Development of Benchmark Environment

II.1 Design Goals and Requirements

Our goal in developing a benchmark system is to provide an environment for continuous benchmark and improvement in optimization tools for 300mm foundry manufacturing services. In specific, the benchmark system should support the closed-loop-control mechanism to efficiently integrate all the up-to-date research results from each subproject and all the insightful feedbacks from system visitors. To fulfil these requirements, the roles and processes involved in a benchmark environment are first analyzed and designed. Then a framework for developing a scalable benchmark system is designed and constructed.

In our design of benchmark system, there are three roles: subproject owner, website manager, and system visitor. According to the interactions between the roles and the benchmark system, three levels of processes are identified: operational, analytic, and collaborative. Fig 2 demonstrates the roles and processes involved in a benchmark environment through a table format, where the major functions/requirements for each combination of a role and a level of process are demonstrated as list items in each cell. The focus of this section is on the process in the operational level. The process in the analytic level will be described in section III. As for the process in the collaborative level, it is not designed yet and will be the focus of next year's research.

II.2 A Framework for Scalable Benchmark Environment Development

To fulfil all the requirements and functions illustrated in Fig 2, we design a framework for scalable benchmark environment development. The framework is designed to support the efficient integration of all the up-to-date research results from each subproject and all the insightful feedbacks from system visitors.

The major idea behind the framework is to enhance the system development efficiency, flexibility and scalability by improving the reusability in system module, component, object, and function levels. Fig 3 demonstrates our design of framework using this idea, where the concept of the most advanced service-oriented architecture (SOA) [2] is applied here. It can be seen that five kinds of services are identified and developed: UI Tool, 300mm Tool, Authority, DB and File services. All these five services are loosely coupled with each other and the system, i.e. the functions providing by these services are independent on the contents sending or requested from the service consumers [3].

In addition, special designs are conducted on the WEB server and AP server. In the WEB server, the MVC (model-view-control) design pattern [4] is applied so that there are clear separations among UI contents layout, UI functional-link controller, and the data and functions in AP server. In this way, it not only enables the concurrent development of UI page, UI controller and AP server but also greatly enhance the system reusability and configurability.

As for the AP server, the Object-Oriented design [5] is applied. On one hand, for the easy integration of all resources on the benchmark system, a generic object is designed and developed to wrap all kinds of data, information and knowledge [6]. The access of all the generic objects is through the object manager, which manipulates the generic object by dynamically accessing the object definition. On the other hand, to facilitate the classification and management of all types of generic objects and their instances, a tree node object is designed and inherited from the generic object. The management and query of all the tree nodes is through the tree manager, which traverse all the tree nodes by dynamically accessing the tree definition.

II.3 Development of Benchmark System in Operational Level

To develop a benchmark system, we should first design its major processes. One of the major processes in the operational level is to create, classify, and view contents on the benchmark system. Fig 4 shows the sequence diagram [7] of this process, where three roles (subproject owner, website manager, system visitor) are involved. Three system modules are then designed to support this major process: project repository, website management and website explorer. The first module, project repository, provides a mechanism for individual subproject members to manage their project data, information, and knowledge on the same web environment. The second module, website management, is designed for the website administrations such as the construction of website structure, management of system users, management of object type definition, etc. The third module, website explorer, facilitates system visitors to explore the benchmark system.

Based on the roles involved in the operational process, the sequence diagram in Fig 4 is separated as three sets of steps. The first set of steps is for the subproject owner, who will manipulate (create, edit, upload, download) their project contents (in text, html, pdf, or microsoft file formats) through the project repository on the benchmark system. The benchmark system is designed to support the subproject owner to sequentially select project folder, create and edit page files, and bind page file with the project folder (Fig 5~7). The second set of steps is for the website manager, who will manage the classification structure nodes and bind them with the page files in project repository for website exploring. The benchmark system is designed to support the website manager to sequentially select/edit website structure node, select page file from project repository, and bind the website structure node with the selected page file (Fig 8~10). The third set of steps is for the system visitor. The benchmark system is designed to facilitate system visitors explore the website following the instructions of website classification structure (Fig. 11).

The analysis and design of the sequence diagram discussed above helps us identify the major objects and functions to realize the benchmark system in operation level, based on which the class diagram [8] is therefore designed. Fig 4 depicts the class diagram of a benchmark system in operational level. It is designed on the top of the framework demonstrated in Fig 3. We can see that three object classes (tree manager, tree node, and generic object) in the previously developed framework are reused and extended for the design and implementation of five object classes (project repository, project folder, website management, website structure node, and page file) in the benchmark system. This illustrates the advantages of using the framework in Fig 3 for benchmark system development.

III. Design and Development of Optimization-Tools Integration

III.1 Architecture for Flexible Optimization-Tools Integration

To perform simulation and analytic computing service, we design a function module to hookup developed analytic tools and Benchmark system. (see Fig 13) First, we sort out two types of tool sources. For those who directly shared their developed tools, we integrate them into Benchmark system as server side computing service. Otherwise, since confidential issues may arise when sharing some tools, the web-service [9] components from remote server could be then designed and integrated into the system.

Due to various program developing platforms, we are facing different formats of analytic tools. There are two major issues:

- I1. The analytic tools are shared as execution files (.exe) or dynamic linking libraries (.dll). Also, developers of some tools might just provide web-service components for remote accessing via internet.
- I2. The provided tools or service may require input files for program settings. The formats of input files are surely different. Furthermore, the output file as the analyzed results of each tool would be varied.

To deal with different types and formats of tools, we first design an adaptor (a generic tool adapter), to handle external tools from various sources. Then, specific adapters, ex: EXE Adapter and DLL Adapter, inherited from Adapter are designed to integrate specific types of tools into Benchmark system.

III.2 An Example of Optimization-Tools Integration

Here, we use an example to demonstrate the mechanism of optimization-tools integration. Assuming Dr. Hu from NTU provides a discrete event dynamic system (DEDS) simulator [10] to be integrated into Benchmark system as a web-based computing service. This tool is shared as an execution file (.exe) which is developed in C language and compiled by Visual C++ 6.0. It also needs to be set up by specific input settings defined in a text file. After running the program, simulation statistics and results are written in an output file. The standard procedures to integrate this tool into the system are as follows:

a. Collect the execution file (includes run-time components) and input files. Upload these files to the designated directory by using the project repository management on Benchmark website.
b. To execute the DEDS simulator on the Benchmark web page, a “deds.php” file is created by using the EXE Adapter. The designed page should perform four functions:
    (1) Allow clients to upload modified input files.
    (2) Provide a user interface where the parameters for various scenarios and strategies could be manually set by clients.
    (3) Execute the tool based on the given input settings and parameters.
    (4) Display the simulation results and http link for downloading the output files.
c. We then establish a classification node in Web Structure and connect it to the designed page (“deds.php”). Therefore clients could access and use the computing service on this page via internet.
d. Finally, we demonstrate this tool from client-side and test if it works well. As can be seen in Fig 14, the page “deds.php” contains all necessary functions. After execution, Fig 15 shows up a page with simulation results. The output file is also downloadable on the page.

III.3 Summary of Optimization-Tools Integration

To achieve the goal of benchmarking developed tools, we design the integrating mechanism, which includes extendable adapters and standard procedures, to integrate external analytic tools into Benchmark system. As can be seen in Fig 16, there might be more varied sources of tools in the future. Follow the procedure described in section III.2, we could build a extendable web-based test board to integrate more varied analytic tools and provide overall computing service for benchmarking.

IV. Optimization Tools Developed by Individual Subprojects (2nd Year)

Subproject 1: a collaborative scheduling tool for semiconductor supply chain

This project proposes to develop enabling technologies for collaborative planning and scheduling in semiconductor supply chains. In the second year, our progress includes a study of system dynamics of production operation, completion of scheduling tools for the chains, and an analysis of the behavior of material flow in the chains.

Subproject 2: Ordinal optimization-based policy iteration algorithm for dynamic schedule composition

In the second year, we have completed the design, and convergence analysis of an OO-Based Policy Iteration (OOBPI) method to handle the combinatorial complexity of decisions over the time axis for dynamic composition of production schedules. Utilizing the framework of policy iteration, we approximate the optimal cost-to-go and optimal decision of each state by simulation-based OO. Priority service disciplines are also included into the simulator.

Subproject 3: Dispatching policy and simulator for differentiated material handling services

In the second-year research, we accomplished three designs:

(1) a differentiated preemptive dispatching policy to expedite the movement of high priority products to the extreme while keeping the incurred delays on regular ones being acceptable;

(2) an optimization-based vehicle control system for differentiated 300mm AMHS with the help of integer programming techniques;

(3) Lagrangian relaxation and Gauss-Seidel iteration methods for shop floor control of hot lots by allocating proper quota of each priority level of products such that the overall throughputs are maximized while the specified cycle time performances of each priority level are satisfied.

Subproject 4: **An evolutionary process optimization method for multi-stage processes**

In the second year, the proposed evolutionary optimization method in the first year is further enhanced for a multi-stage manufacturing system. Focus is on the interactions among steps and how parameters of a local processing step should be designed such that the final system output is improved. We proposed an efficient method to approximate confidence intervals of the process response surface (Fig 19). The proposed confidence interval approximation can be then used to facilitate evolutionary process improvement. By using the confidence interval minimization as a design criterion, a sequential experimental design procedure (Figure 20) is proposed to find the near-optimum point for multi-step processes.

V. Conclusions and Future Developments

The major processes, roles, and requirements of a benchmark environment for 300mm manufacturing services have been identified. Up to now, we have designed and developed a flexible and scalable benchmark system, which is on top of two achievements: constructions of the architecture for flexible optimization-tools integration and a framework for scalable benchmark-environment development. The developed benchmark system is able to support the processes for subproject owners, website manager, and system visitors in operational and analytic levels.

Based on the developments and outcomes in the first two years, the third year will focus on the following tasks:

T1. Hookup all the latest tools developed in each subproject

Currently, the methodologies and tools in each subproject are still under development. We will use the architecture for flexible optimization-tools integration as a base to efficiently hookup all these tools.

T2. Design a specific benchmark analysis mechanism for each optimization problem

In addition to the integration of heterogeneous optimization-tools, we also need a benchmark-analysis mechanism specifically designed for the salient features of individual optimization problems.

T3. Design and development of a collaborative process for continuous improvement

Our ultimate goal is to realize an environment for continuous benchmark and improvement in optimization tools for 300mm manufacturing services. It needs a mechanism to manage all the benchmark-system issues for the direction, monitoring, and control of system improvements. For this purpose, we will design and development of a collaborative process for issue tracking among system visitors, subproject owners, and website manager.

三、參考文獻

[1] http://140.112.37.24/benchmark/ui_user_view.php

[2] O. Zimmermann et al, “Elements of Service-Oriented Analysis and Design”, http://www-128.ibm.com/eveloperworks/webservices/library/ws-soad1/

[3] A. Arsanjani, “Service-Oriented Modeling and Architecture”, www.ibm.com/developerworks/webservices/library/ws-soa-design1/

[4] F. Buschmann et al, Pattern-Oriented Software Architecture, Volume 1: A System of Patterns, John Wiley & Sons, 2001.

[5] G. Booch, Object-Oriented Analysis and Design with Applications, Addison-Wesley Professional, 1993.

[6] Resource Description Framework (RDF), http://www.w3.org/ RDF

[7] A. Dennis, Systems Analysis and Design with UML Version 2.0 : An Object-Oriented Approach, John Wiley & Sons, 2004

[8] H. Eriksson, UML 2 Toolkit, Wiley,2003

[9] A. Ryman, “Understanding Web Services”, http://www-128.ibm.com/developerworks/websphere/library/techarticles/0307_ryman/ryman.html

[10] M. Hu, Translating Overall Production Goals into Distributed Flow Control Specifications for Semiconductor Wafer Fabrication, Ph.D. Thesis, Institute of Electrical Engineering, National Taiwan University, 2002.

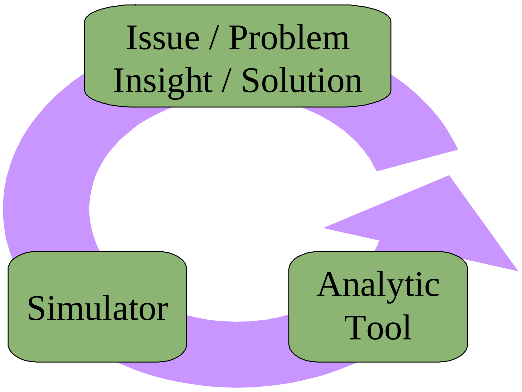
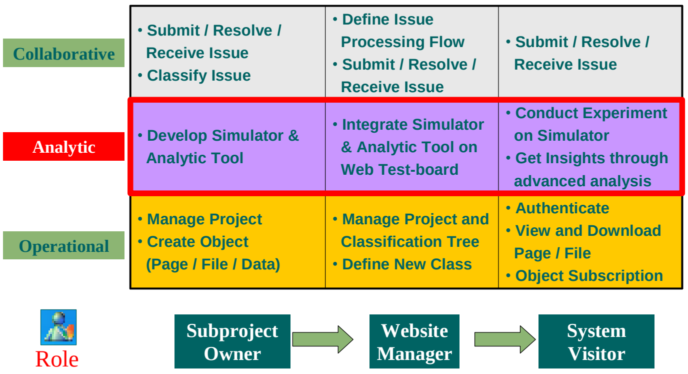


Fig 1: Continuous benchmark and improvement in optimization tools for 300mm foundry manufacturing services



Role	Subproject Owner	Website Manager	System Visitor
Collaborative	• Submit / Resolve / Receive Issue • Classify Issue	• Define Issue Processing Flow • Submit / Resolve / Receive Issue	• Submit / Resolve / Receive Issue
Analytic	• Develop Simulator & Analytic Tool	• Integrate Simulator & Analytic Tool on Web Test-board	• Conduct Experiment on Simulator • Get Insights through advanced analysis
Operational	• Manage Project • Create Object (Page / File / Data)	• Manage Project and Classification Tree • Define New Class	• Authenticate • View and Download Page / File • Object Subscription

Fig 2: The roles, processes, and functions involved in a benchmark environment

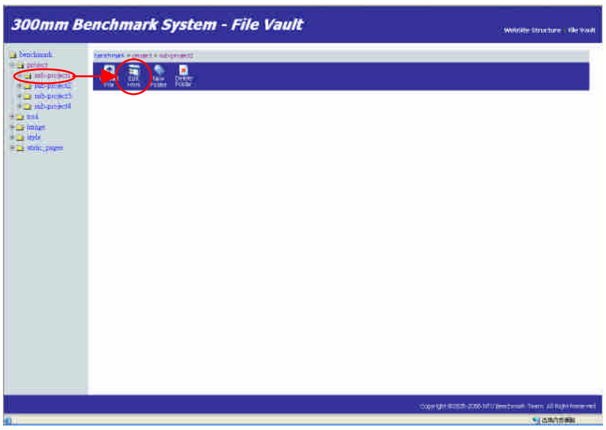


Fig 5: The environment for project repository management

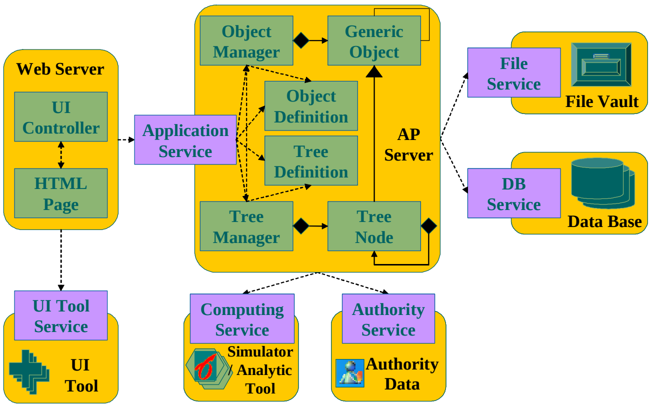
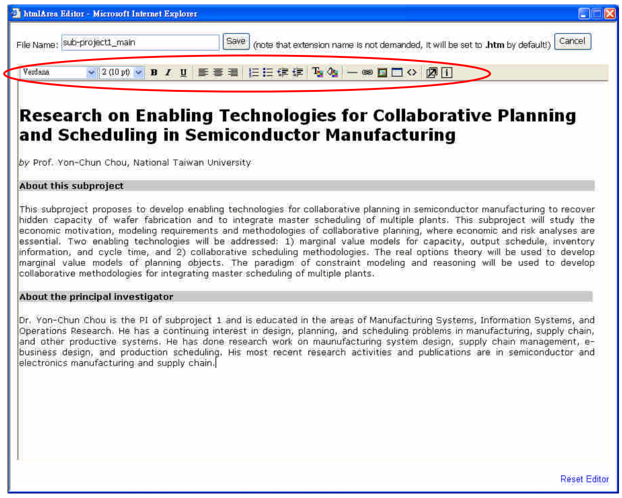


Fig 3: The framework for scalable benchmark environment development



File Name: sub-project1_main (note that extension name is not demanded, it will be set to **.htm** by default!)

Research on Enabling Technologies for Collaborative Planning and Scheduling in Semiconductor Manufacturing

by Prof. Yon-Chun Chou, National Taiwan University

About this subproject

This subproject proposes to develop enabling technologies for collaborative planning in semiconductor manufacturing to recover hidden capacity of wafer fabrication and to integrate master scheduling of multiple plants. This subproject will study the economic motivation, modeling requirements and methodologies of collaborative planning, where economic and risk analyses are essential. Two enabling technologies will be addressed: 1) marginal value models for capacity, output schedule, inventory information, and cycle time, and 2) collaborative scheduling methodologies. The real options theory will be used to develop marginal value models of planning objects. The paradigm of constraint modeling and reasoning will be used to develop collaborative methodologies for integrating master scheduling of multiple plants.

About the principal investigator

Dr. Yon-Chun Chou is the PI of subproject 1 and is educated in the areas of Manufacturing Systems, Information Systems, and Operations Research. He has a continuing interest in design, planning, and scheduling problems in manufacturing, supply chain, and other productive systems. He has done research work on maunufacturing system design, supply chain management, e-business design, and production scheduling. His most recent research activities and publications are in semiconductor and electronics manufacturing and supply chain.

Fig 6: The web-based customized page editor

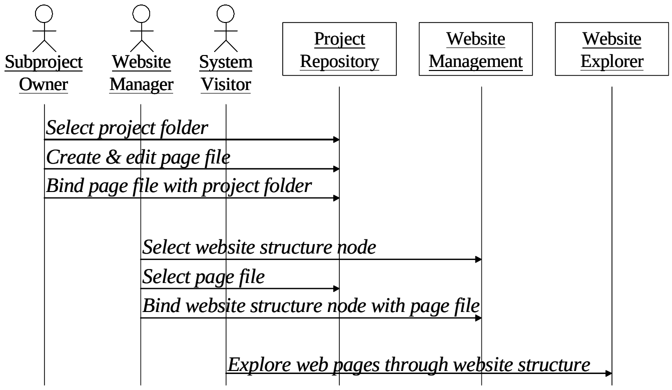
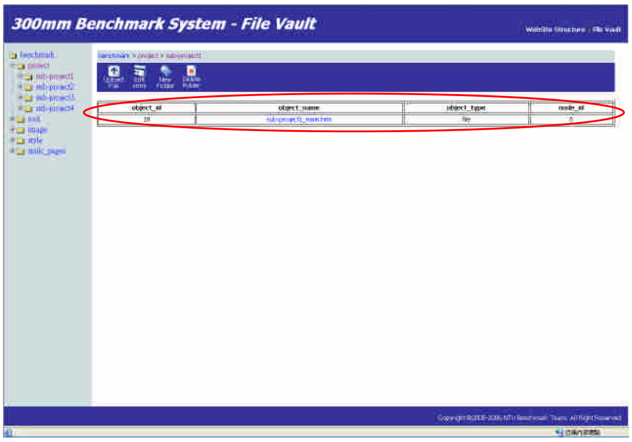
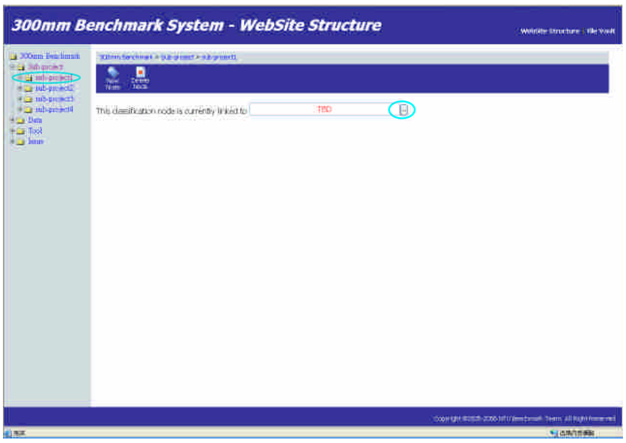


Fig 4: An operational process to create, classify and view contents



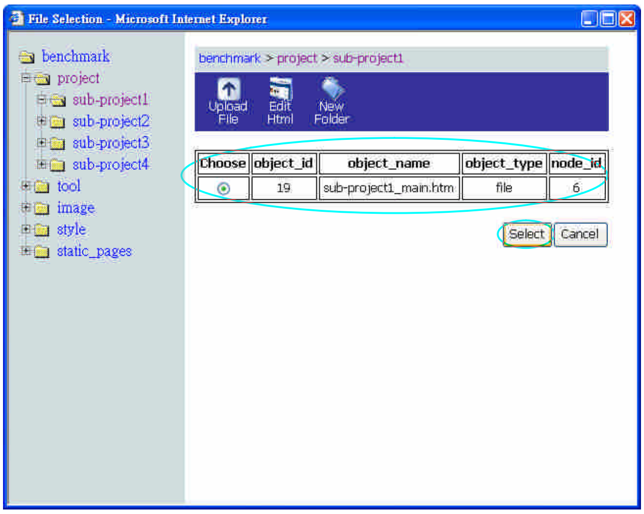
object_id	object_name	object_type	node_id
19	sub-project1_main.htm	file	[illegible]

Fig 7: The binding of project repository folder and customized page



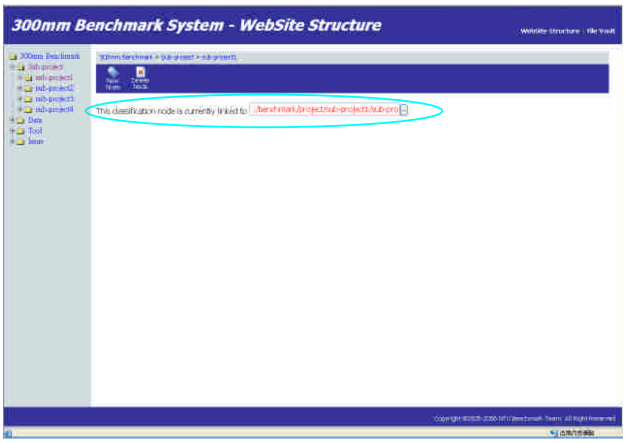
This classification node is currently linked to TBD

Fig 8: The environment for website structure management



Choose	object_id	object_name	object_type	node_id
●	19	sub-project1_main.htm	file	6

Fig 9: Selection of a page-file to bind with a website structure node



This classification node is currently linked to ../benchmark/project/sub-project1/sub-pro

Fig 10: Website structure after binding

Research on Enabling Technologies for Collaborative Planning and Scheduling in Semiconductor Manufacturing

About this subproject

About the principal investigator

Fig 11 System user view page on the website explorer

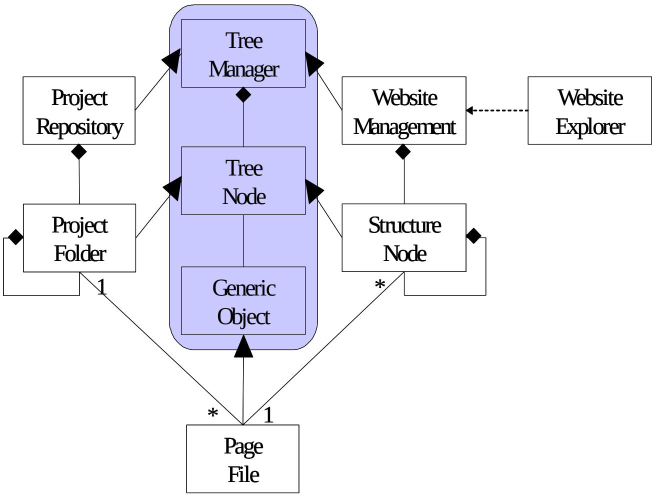


Fig. 12: The class diagram of a benchmark system

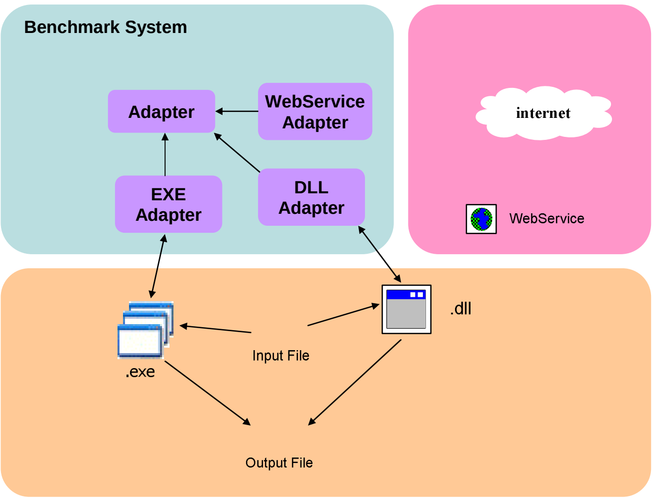
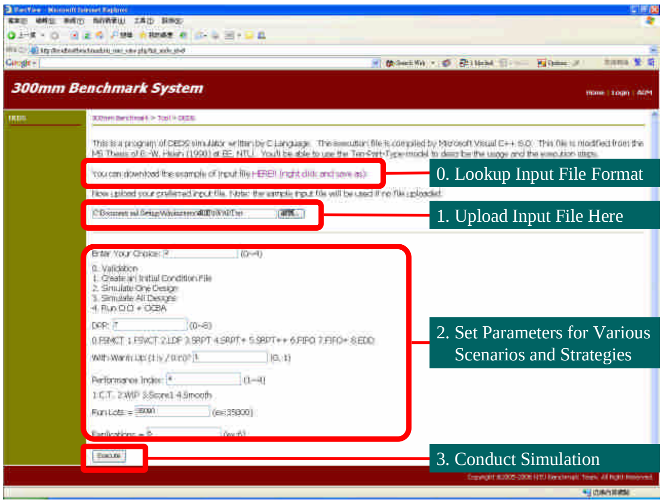


Figure 13: Computing Service Module for Developed Analytic



This is a program of DEDS simulator written by C Language. The execution file is compiled by Microsoft Visual C++ 6.0. This file is modified from the MS Thesis of E.-W. Hsieh (1990) at IE, NTU. You'll be able to use the Ten-Part-Type-model to describe the usage and the execution steps.

You can download the example of input file HERE! (right click and save as)

0. Lookup Input File Format

Now upload your preferred input file. Note: the sample input file will be used if no file uploaded.

1. Upload Input File Here

Enter Your Choice: 2 (0~4)

0. Validation
1. Create an Initial Condition File
2. Simulate One Design
3. Simulate All Designs
4. Run OO + OCBA

DPR: 7 (0~8)

0.FSMCT 1.FSVCT 2.LDF 3.SRPT 4.SRPT+ 5.SRPT++ 6.FIFO 7.FIFO+ 8.EDD

With Warm Up (1 is / 0 is no)? 1 (0, 1)

Performance Index: 4 (1~4)

1.C.T. 2.WIP 3.Score1 4.Smooth

Run Lots = 35000 (ex:35000)

Replications = 5 (ex:5)

2. Set Parameters for Various Scenarios and Strategies

3. Conduct Simulation

Fig 14 "deds.php" demonstration

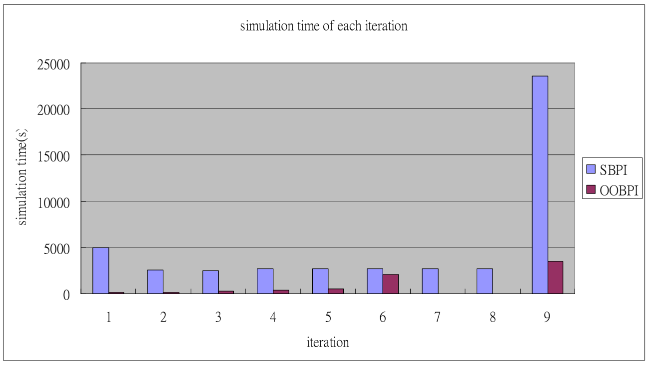
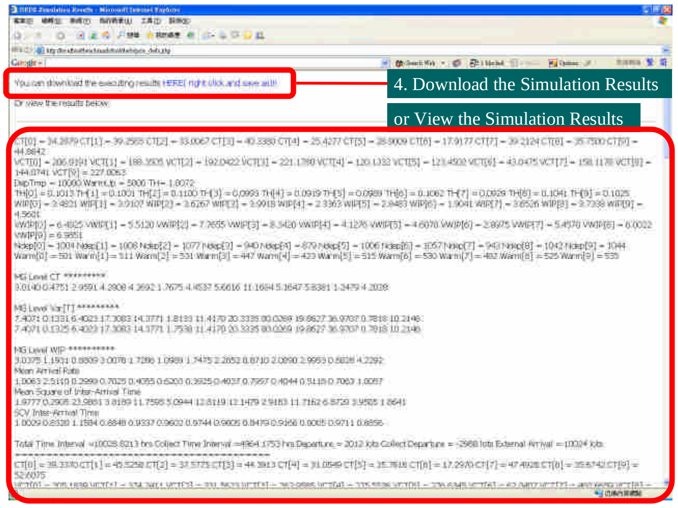


Fig 17: Simulation time of OOBPI vs. SBPI



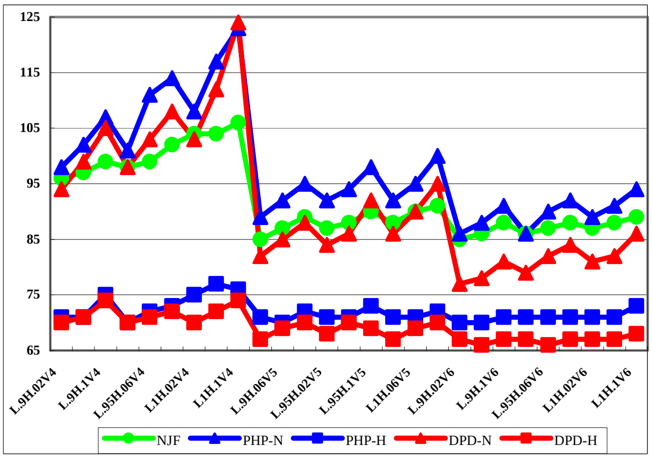
You can download the executing results HERE(right click and save as)!!

4. Download the Simulation Results

Or view the results below

or View the Simulation Results

Fig 15 Simulation results and downloadable output file



1. PHP-H is PHP for hot lots and PHP-N is PHP for normal lots.
2. DPD-H is DPD for hot lots and DPD-N is DPD for normal lots.
3. L indicates bay loading, e.g., L9 implies 90% loadings in the loop.
4. H indicates hot lots percentage, e.g., H.02 implies 2% of hot lots.
5. V indicates OHT vehicle number, e.g., V4 implies 4 OHT vehicles.

Figure 18. Simulation average results in different configurations (in seconds)

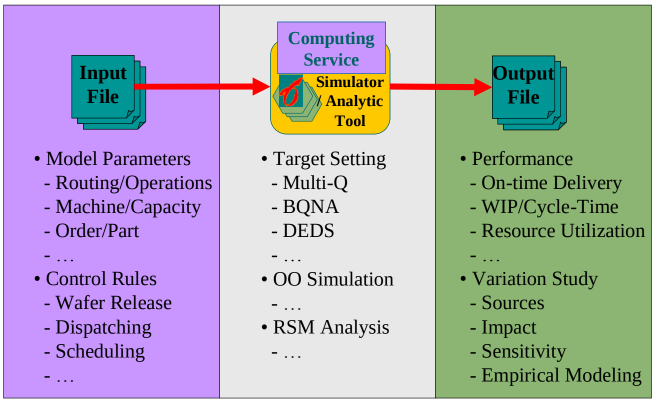


Fig 16 Future web-based test board

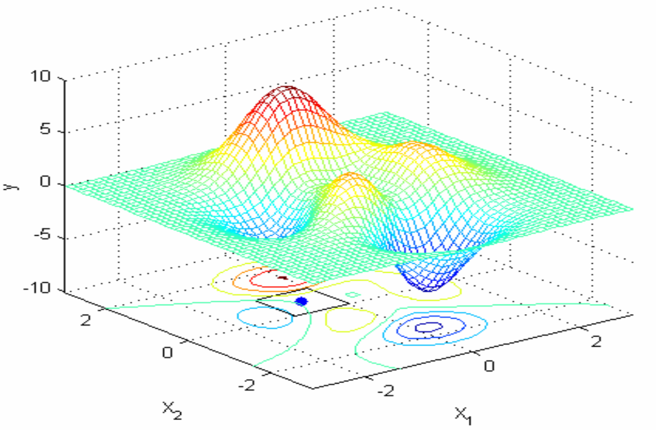
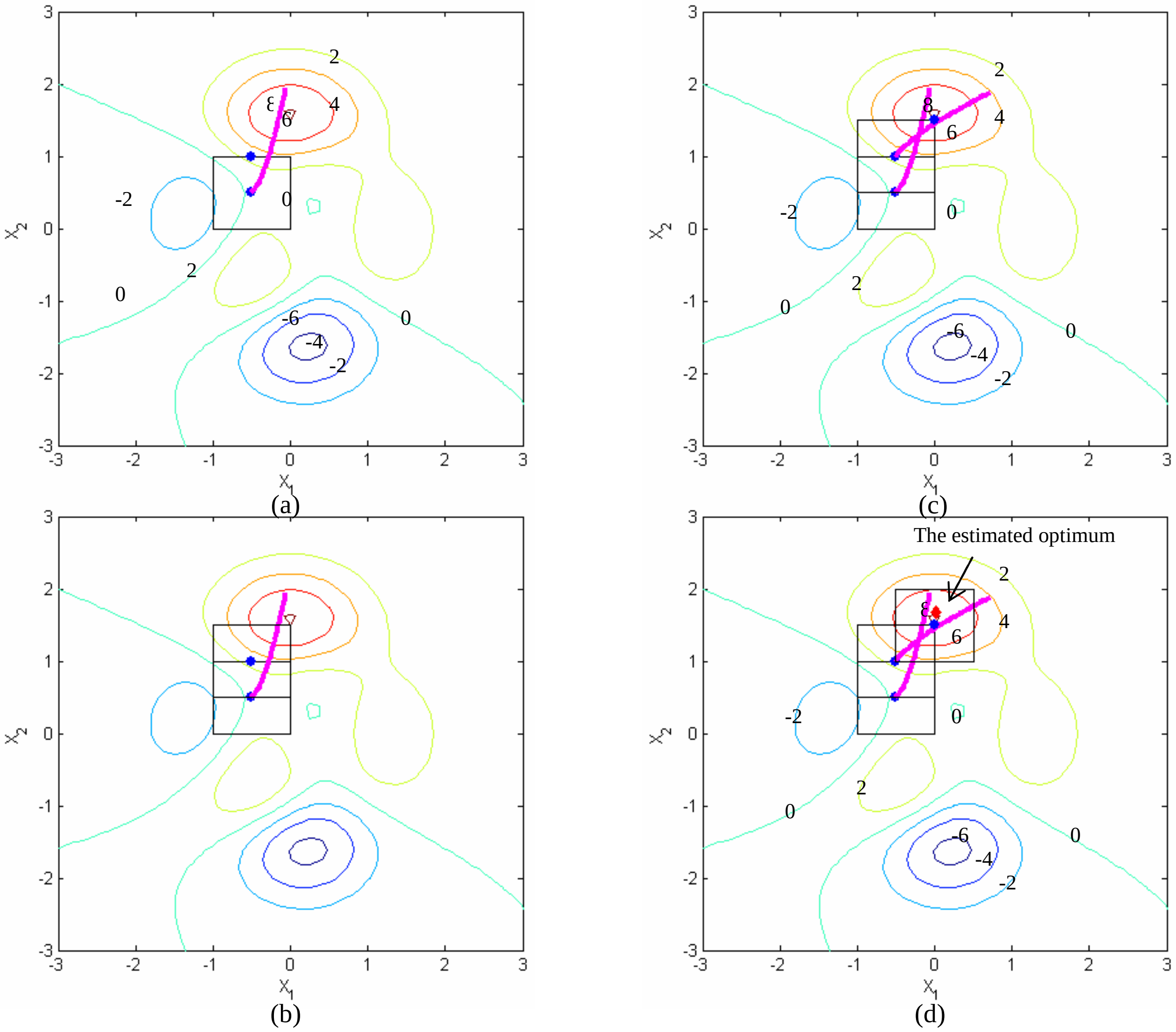


Figure 19: Process response surface



(a) (c)

(b) (d)

Fig 20: The contour plots of sequential steps for process improvement.