

逢 甲 大 學  
交通工程與管理學系碩士班  
碩 士 論 文



供應鏈管理下需求預測控制之研究  
A Study of Demand Predictive Control  
Model for Supply Chain Management

指導教授：胡大瀛

研 究 生：蘇怡如

中 華 民 國 九 十 四 年 七 月

## 誌 謝

求學生涯中，影響我最深的莫過於是我的指導教授 胡大瀛教授。感謝老師引領我進入研究的領域，並帶給我許多自信，讓我能更積極的面對事情，無論在課業以及待人處世上受益不少。老師的諄諄教誨，我會銘記於心，在此獻上學生最深的感謝之意。

論文口試期間，承蒙中央大學 顏上堯教授與海洋大學 顏進儒教授在百忙之中抽空指導，提供許多寶貴意見，使學生的論文更加完整。此外，在論文撰寫期間承蒙系上 李克聰教授與 林大傑教授細心審閱與謬誤之指正，給予許多啟發與關心。另外也感謝系上其他老師與三位助教在生活上或學業上的照顧與幫助。

論文期間，尤其感謝同窗好友炯男與德政的幫忙，很幸運能與你們一起共事。感謝光鎧、俊宏、文隆、品宏、秋評、怡芳、裕雯學姊、冠樺、彥斐、宇軒、曜彰、品翰、自強、雲慶、偉強、孟宗及宗憲在生活及課業上的互相關懷勉勵。以及其它親愛的同學們，很開心與你們相聚兩年，我會好好珍惜這段緣份的。

在逢甲六年，完成了我的學士及碩士學位，感謝達瑾學長的提攜，讓我有幸於大三進入像大家庭的 Pluto。感謝景文學長在專題過程中不厭其煩的指導；蔚明學長與建緯學長在生活及課業上的意見提供；百賢學長在論文程式上的協助。感謝麗雯學姊的指導以及關心，讓我學到了許多做事做人的道理以及不同的思考方式，受用不少。可愛的學弟妹錦昌、于司、庭銀、彥臻以及大昕，謝謝你們讓我有不少的收穫。還有感謝大學死黨們以及摯友子豪，你們的傾聽與關心是我重要的支柱之一。

最後謹將此份論文獻給我最親愛的家人，感謝父母給我最好的學習環境，以及同在台中求學的妹妹能與我相互扶持，讓我在低潮時始終有個人陪伴我。你們的支持鼓勵與關心是我精神上的後盾，我會更加努力，謝謝你們！

謹誌

2005 年 仲夏

## 摘要

在全球化競爭的情況下，商業交易環境已從以往企業與企業間的競爭演變到不同供應鏈體系間的競爭。如今供應鏈管理(Supply Chain Management, SCM)成為各企業在經營管理上的一個重要課題，主要是將企業間各功能部門間相關的運作程序加以整合協調之合作策略，其涵蓋範圍可從物料取得，經過設計、生產、庫存管理、行銷與售後服務等。供應鏈在變動環境中，存在著許多不確定性，其中需求不確定性即是由客戶的需求變異所產生，為許多企業所欲掌握的一個重要部分，因此需求預測為一重要因素。近年來，許多研究開始應用模式預測控制(Model Predictive Control, MPC)在存貨控制或供應鏈管理上，與傳統最佳化成本隨機規劃法(cost-optimal stochastic programming solutions)相比，所需之技巧較為簡易，也不須建構在許多假設下；在資訊分享、路網架構以及限制處理上皆較具彈性。因此本研究欲發展一模式預測控制模式，模式預測控制為一控制理論，包含決策變數的更新以及需求預測模式，以能在相關因素變動時尋找最佳決策變數，進行決策。

本研究主要構建動態供應鏈管理模式，並將模式預測控制應用於供應鏈管理，首先定義供應鏈組成架構，並分析架構階層間之關係及限制，依此構建供應鏈模式。接著構建模式預測控制，包含決策變數及需求預測，由於過去及現在的控制行為會影響未來系統的回應結果，而滾動平面法(rolling horizon approach)可藉由結合過去及現在的控制行為預測未來之控制行為，因此選用此一方法進行控制行為，在系統因素或環境有所變動時進行決策更新。此外，構建一自我迴歸移動平均整合(Auto Regressive Integrated Moving Average, ARIMA)時間序列模型進行需求預測。

實驗案例設計上分為兩部份，首先建立需求預測模型，再進行最佳化求解。在需求預測模型之建立上，其實驗數據來源以實際訂單數列資料校估獲得預測模型參數。供應鏈所需資料則由不同型態之虛擬資料進行最佳化以獲得決策，以驗證模型。期望能完成基礎之模式預測控制模型，以能在不同產業下彈性修正模型，提供產業決策建議。

關鍵詞：供應鏈管理、需求預測、模式預測控制

## Abstract

In the global competition, Supply Chain Management (SCM) plays an important role in enterprise operations and management. SCM is developed to coordinate organizational operations, which encompassing raw materials, manufacturing, sales and service to serve customers. There are lots of uncertainties exist in the Supply chain due to the violent market environment. Among these uncertainties, demand uncertainty is originated from the customer's demand. Therefore, demand forecasting is an important factor in SCM. Recently, lots of researches focus on applying Model Predictive Control (MPC) in inventory control or SCM. The MPC approach provides the flexibility in information share, network structure, and constraint processing. This research proposes a MPC model to model the demand forecasting problems with updating process for decision variables.

(the supply chain structure) This research constructs a dynamic SCM model, and applies MPC to the supply chain management. The dynamic SCM model is constructed based on a proposed supply chain structure, and elements in the SCM are analyzed to explore the relationship and restriction of variables among the echelon structure. In demand forecasting, the Auto Regressive Integrated Moving Average (ARIMA) model is implemented to forecast future demands based on historical sale records. The MPC includes decision variable and demand forecasting models. Based on the past and present control actions, a rolling horizon approach is proposed to predict future control strategies. In addition, construct for demand forecasting.

In order to illustrate the MPC model, two case studies are conducted, namely, demand forecasting model and optimization model. The demand forecasting model is calibrated based on the actual data. The proposed supply chain system is verified by numerical analysis through a set of simulated data.

**Keywords:** supply chain management 、 model predictive control 、 demand forecasting.

## 目錄

誌謝 .....	I
中文摘要 .....	II
英文摘要 .....	III
目錄 .....	IV
圖目錄 .....	VI
表目錄 .....	VIII

### 第一章 緒論

1.1 研究背景與動機 .....	1
1.2 研究目的 .....	2
1.3 研究方法與流程 .....	3

### 第二章 文獻回顧

2.1 供應鏈 .....	5
2.2 供應鏈問題描述 .....	7
2.3 供應鏈管理的求解方法 .....	8
2.4 供應鏈管理之不確定性因素 .....	11
2.5 時間序列型的預測方法 .....	14
2.5.1 移動平均法 .....	15
2.5.2 指數平滑法 .....	16
2.5.3 迴歸分析法 .....	16
2.5.4 自我迴歸移動平均整合法 .....	16
2.6 模式預測控制 .....	20
2.7 滾動平面法 .....	23

### 第三章 供應鏈模型架構之建立

3.1 供應鏈模式 .....	25
3.1.1 定義供應鏈問題 .....	26
3.1.2 模式構建 .....	28
3.2 模式預測控制架構 .....	33
3.3 需求預測模式 .....	35

3.4 最佳化模式.....	36
第四章 模式求解演算法之建立與探討	
4.1 整體模式求解架構.....	39
4.2 需求預測.....	40
4.3 最佳化求解.....	46
第五章 數值實驗	
5.1 實驗設計.....	48
5.1.1 背景介紹.....	48
5.1.2 實驗設計.....	50
5.1.3 實驗資料.....	53
5.2 多階層單一產品實驗.....	56
5.2.1 需求預測模型.....	56
5.2.2 最佳化.....	60
5.2.3 結果討論.....	72
5.3 多階層多產品實驗.....	73
5.3.1 需求預測模型.....	73
5.3.2 最佳化.....	80
5.3.3 結果討論.....	94
第六章 結論與建議	
6.1 結論.....	96
6.2 建議.....	97
參考文獻.....	98
附錄	
附錄一 自行車出口數量及出口單價資料.....	101
附錄二 多階層多產品最佳化結果.....	106
附錄三 應用CPLEX求解模式程式.....	115

## 圖目錄

圖 1.1	研究流程圖 .....	4
圖 2.1	傳統供應鏈相對資訊供應鏈示意圖 .....	6
圖 2.2	供應鏈架構圖 .....	7
圖 2.3	供應鏈管理之不確定性因素的影響 .....	12
圖 2.4	ARIMA 標準流程圖 .....	17
圖 2.5	模式預測控制架構 .....	21
圖 2.6	滾動平面法示意圖 .....	24
圖 3.1	供應鏈路網圖 .....	26
圖 3.2	MPC 策略架構 .....	33
圖 3.3	模式預測控制概念圖 .....	34
圖 3.4	滾動平面法下決策更新流程 .....	35
圖 4.1	演算流程圖 .....	40
圖 4.2	需求預測模式產生流程圖 .....	42
圖 4.3	CPLEX 最佳化求解流程 .....	47
圖 5.1	自行車每月出口總數量 .....	49
圖 5.2	自行車每月出口單價 .....	50
圖 5.3	自行車每月出口銷售額 .....	50
圖 5.4	國內自行車產業上、中、下游關聯圖 .....	51
圖 5.5	實驗案例架構 .....	52
圖 5.6	不同歷史資料筆數之需求預測案例 .....	53
圖 5.7	多階層單一產品供應鏈路網 .....	56
圖 5.8	d=0、1 下數列資料之 ACF 圖 .....	57
圖 5.9	數列資料之 PACF 圖 .....	57
圖 5.10	(p, d, q)=(1, 1, 1)殘差之 ACF 圖、PACF 圖 .....	58
圖 5.11	需求預測模型預測結果 .....	60
圖 5.12	程式示意圖 .....	62
圖 5.13	CPLEX 求解結果示意圖 .....	62
圖 5.14	決策結果路網圖(4 個時段) .....	63
圖 5.15	決策結果路網圖(4 個時段.情境一) .....	65
圖 5.16	決策結果路網圖(4 個時段.情境二) .....	66
圖 5.17	決策結果路網圖(10 個時段) .....	68

圖 5.18	決策結果路網圖(10 個時段.情境一).....	70
圖 5.19	決策結果路網圖(10 個時段.情境二).....	71
圖 5.20	自行車出口至北美自由貿易區預測資料與實際資料趨勢圖 .....	74
圖 5.21	自行車出口至歐盟預測資料與實際資料趨勢圖 .....	75
圖 5.22	自行車出口至歐協預測資料與實際資料趨勢圖 .....	76
圖 5.23	電動自行車出口至北美自由貿易區預測資料與實際資料趨勢圖 .....	77
圖 5.24	電動自行車出口至歐盟預測資料與實際資料趨勢圖 .....	78
圖 5.25	變數時段說明 .....	81
圖 5.26	最佳化結果(訂單在 $t=0$ ) .....	91
圖 5.27	情境一最佳化結果(訂單在 $t=2$ ) .....	93
圖 5.28	情境二最佳化結果(裝配廠在 $t=0$ ) .....	94





## 表目錄

表 2.1	分散型方法 .....	9
表 2.2	集中型方法 .....	10
表 3.1	變數表 .....	27
表 4.1	AR、MA 及 ARMA 模式判斷標準 .....	43
表 4.2	在 d 次差分下 ACF 及 PACF 之情形 .....	44
表 5.1	供應鏈基礎資料 .....	54
表 5.2	前置時間(物料-零件商) .....	55
表 5.3	前置時間(零件-裝配廠) .....	55
表 5.4	前置時間(產品-經銷商) .....	55
表 5.5	生產零件時間 .....	55
表 5.6	生產產品時間 .....	56
表 5.7	候選模式之 AIC 及 BIC 值 .....	58
表 5.8	殘差自我相關係數 .....	59
表 5.9	線性規劃下最佳化 4 個時段之結果 .....	63
表 5.10	多階層單一產品最佳化目標式結果(4 個時段) .....	67
表 5.11	多階層單一產品最佳化目標式結果(10 個時段) .....	72
表 5.12	自行車出口至北美自由貿易區(NAFTA) .....	74
表 5.13	自行車出口至歐盟(EU) .....	75
表 5.14	自行車出口至歐協(EFTA) .....	76
表 5.15	電動自行車出口至北美自由貿易區(NAFTA) .....	77
表 5.16	電動自行車出口至歐盟(EU) .....	78
表 5.17	出口資料預測之最佳歷史資料筆數 .....	79
表 5.18	變數說明、目標式係數 .....	82
表 5.19	多階層多產品最佳化結果 .....	84

# 第一章 緒論

## 1.1 研究背景與動機

在科技進步、產業環境逐漸改變成全球化競爭的情況下，現今的商業交易環境，已從以往企業與企業間的競爭演變到不同供應鏈體系間的競爭。供應鏈管理 (Supply Chain Management, SCM) 的觀念，是由 Houlihan(1985) 首先提出的概念，所謂的供應鏈管理，簡單來說供應鏈管理乃是將原本獨立之產品作業流程，整合為一整體性以滿足顧客需求為目的之系統。如今供應鏈管理成為各企業在經營管理上的一個重要課題，供應鏈為企業與企業間，將各功能部門間相關的運作程序加以整合協調之合作策略，其涵蓋範圍可以從物料的取得，經過設計、生產、庫存管理、行銷與售後服務等。而供應鏈管理即是將合作策略落實，追求企業合作之效率，以較少之產品前置時間與營運成本之最佳化為考量，是一種企業流程中與其他企業之共同合作之過程，其中包括採購、產品規劃、製造、庫存管理、組裝、倉儲與產品運輸等流程。前者重視的是合作的對象與角色，而後者講究的是方針與方法。

另一方面，從組織架構來看，供應鏈管理則是藉由整合物流、資訊流、資金流等供應路網間的流動，也就是由製造商到最終使用者間的雙向溝通分享動作，產生一具效率及成本效益的流程，以達成即時之交貨及服務，增加企業競爭力。所以供應鏈管理的效益，即要求減少訂單處理次數與降低庫存量來降低成本與無附加價值之活動，加上資訊之傳輸速度與時效、資訊之正確性以及資訊之分享，回應並滿足顧客的需求，改善供應鏈間的溝通效率，完成最大化利益的目標。

供應鏈在變動環境中，存在著許多不確定性，包含需求不確定性、製造不確定性以及供應不確定性，而其中需求不確定性即是由客戶的需求變異所產生，為許多企業所欲掌握的一個重要部分，因此，需求預測為一重要因素。Braun et al., (2003) 近幾年來，許多學者開始

應用模式預測控制 (Model Predictive Control, MPC)在存貨控制或供應鏈管理上，與傳統最佳化成本隨機規劃法 (cost-optimal stochastic programming solutions)相比，為不同概念架構，所需之技巧較為簡易，也不似傳統方法須建構在許多假設下；模式預測控制在資訊分享、路網架構以及限制處理上皆較具彈性。因此，本研究欲發展一模式預測控制，模式預測控制為一控制理論，主要用在求解間斷時間最佳化控制問題，包含決策變數的更新以及需求預測模式，以能在相關因素變動時尋找最佳決策變數，進行決策。

此外，針對需求預測部分，在傳統供應鏈管理中，由於僅有零售商能直接觀察到顧客需求，供應鏈中其他階層僅能以下游訂單進行決策，因而時常出現上游廠商在資訊不足或不明確的情形行下，發生需求放大的問題，因此本研究假設在資訊科技成熟的前提下，供應鏈中各階層均能藉由資訊技術直接以終端客戶的訂單資料作為需求預測之依據，以加強需求預測之準確度。

## 1.2 研究目的

本研究欲構建一動態模型描述供應鏈中各階層個別及相關行為，以及將模式預測控制應用在供應鏈管理上，以強化供應鏈管理效率。本研究的目的可以分為以下五點：

1. 分析供應鏈其組成及組成間之相關性，以集中型管理法之概念構建一整體性的供應鏈模式。
2. 針對需求之不確定性，於模式預測控制中利用 ARIMA 時間序列模型構建一需求預測模式。
3. 利用模式預測控制概念分析供應鏈管理中之決策與控制變數，以有效最大化效益。
4. 利用虛擬之數據資料檢視所構建完成之模式，並予以修正之。

### 1.3 研究方法與流程

本研究主要構建一動態供應鏈管理模式，並將模式預測控制應用在供應鏈管理上，首先定義供應鏈之組成架構，並分析架構階層間之關係及限制，依此構建供應鏈模式。接著構建模式預測控制，包含決策變數及需求預測，選用此一方法進行控制行為，在系統因素或環境有所變動時進行決策更新。此外，構建一自我迴歸移動平均整合 (Auto Regressive Integrated Moving Average, ARIMA) 時間序列模型進行需求預測。

其研究流程如圖 1.1 所示，主要內容及進行流程如下所示：

#### 1. 文獻回顧

除針對供應鏈及供應鏈管理求解法外，還針對模式控制預測、滾動平面法，及自我迴歸移動平均整合法進行回顧。

#### 2. 供應鏈模式及模式預測控制

構件供應鏈模式及模式預測控制模式。

#### 3. 程式撰寫

在系統架構以及模式構建完成之後，著手進程式撰寫。

#### 4. 案例測試分析

研究中將以模擬資料進行測試，以檢驗模式效益。

#### 5. 結論與建議

根據測試分析結果，提出具體的結論與後續研究之建議。

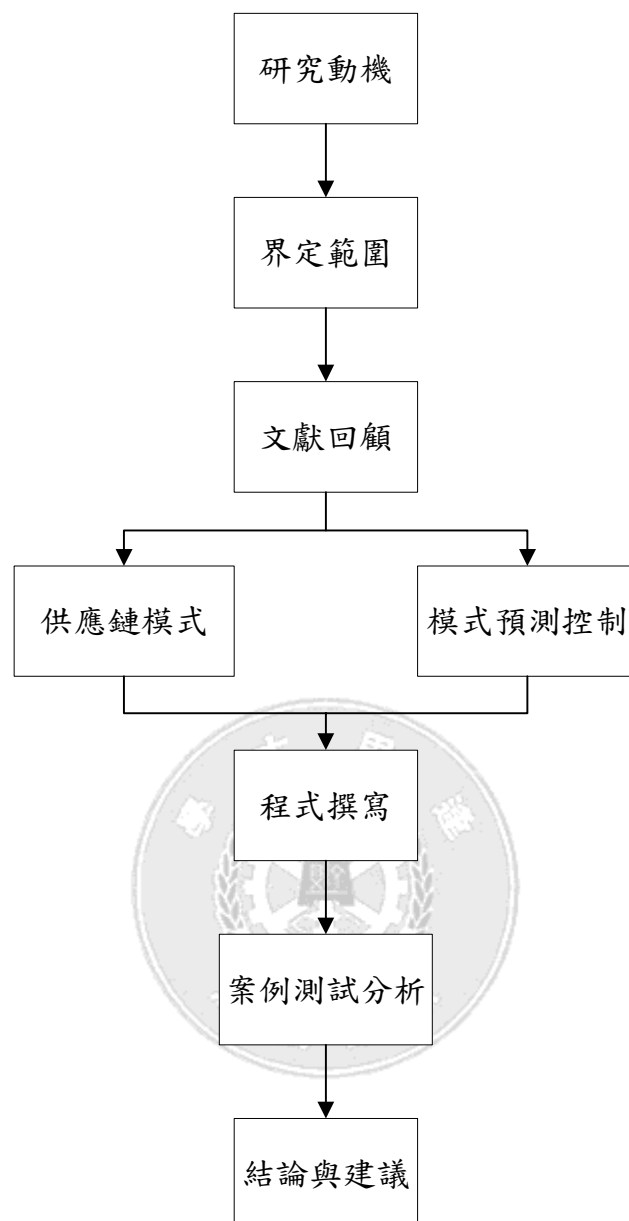


圖 1.1 研究流程

## 第二章 文獻回顧

在本章中將在各節中回顧以下課題，第 2.1 節回顧供應鏈的組織及管理目的、第 2.2 節回顧供應鏈問題中所會發生，並待求解的問題、第 2.3 節回顧供應鏈管理對於其中所發生之問題的求解方法、第 2.4 節回顧供應鏈管理之不確定性因素，以及解決需求不確定因素的相關策略、第 2.5 節回顧時間序列型的預測方法，包含移動平均法、指數平滑法以及自我迴歸移動平均整合法、第 2.6 節回顧模式預測控制、第 2.7 節回顧模式預測控制所會應用到的滾動平面法。

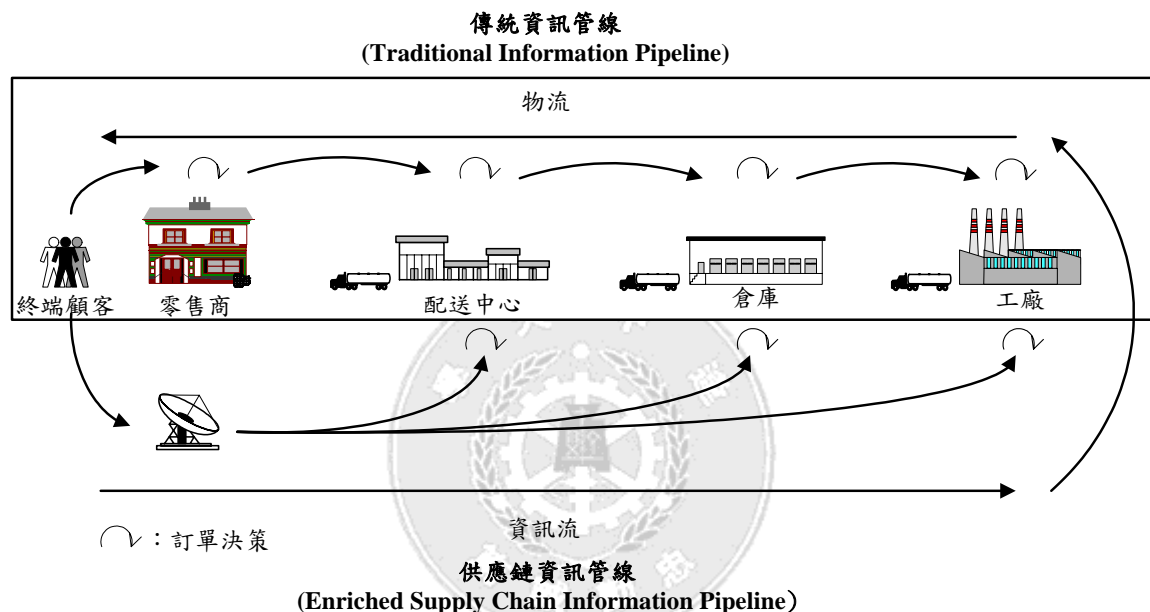
### 2.1 供應鏈

供應鏈管理是以顧客為導向，強調上中下游以專業分工方式，整合企業間所有的價值活動、資訊及資源，來達到虛擬企業的觀念，而供應鏈是由數個階層(echelon)所組成，這些階層直接或間接的滿足了顧客的需求，階層中除了經銷商及製造商之外，還包含運送者、倉庫、零售商以及顧客。這些階層的活動項目包含產品開發、交易、營運、配送、財物以及顧客服務。

一般而言，一個典型的供應鏈通常包含了以下階層 (Chopra and Meindl, 2001)：物料供應商、製造商、經銷商、零售商、顧客。但並非每一個階層均會出現在供應鏈中，必需視顧客需求與產業經營設計，如某些公司，在其供應鏈中是不存在零售商、經銷商或是配送商，因為顧客是直接與公司的製造商下訂單取得貨品的；因此在規劃供應鏈組成階層時，需考慮顧客需求及產業特性。

供應鏈的目標是最大化整體價值，又稱為供應鏈收益性，包含全部供應鏈階層的全部利潤，主要取決於訂單所產生的收益以及供應鏈中所產生的成本，成本因素包含資訊傳遞、生產零件、倉儲、運輸、資金轉換等。供應鏈收益性越大表示此一供應鏈越成功。

供應鏈管理是管理供應鏈中各階層間的物流、資訊流以及資金流，以最大化總利潤。如圖 2.1 所示，在傳統供應鏈中，每一階層在進行訂單決策時，只能依據自身所接到之訂單進行需求預測等相關分析；但在資訊技術發展下，供應鏈中各階層可直接獲得終端顧客的訂單資料進行分析預測，這意味著資訊的傳遞不再是一層一層的往上游傳遞，而是以平行的方式直接傳送給供應鏈中各階層，如此一來可減少需求預測的重複性，降低變異放大的情形發生。



資料來源：Dejonckheere et al., 2003

因此供應鏈管理需有一些決策行為針對這些流動管理之，決策階段可分成規劃、設計以及營運。設計階段主要是決定供應鏈架構以及策略決策(如區位、容量及設備)；規劃階段將進行未來幾年的需求預測以定義出營運方針進行短期營運，供應鏈結構則在此階段中固定成形；營運階段的週期性是一週或每天，此階段中的目標是在已建立的架構及規劃方針下，依據個別顧客訂單進行決策，減低不確定性以及最佳化成果。

## 2.2 供應鏈問題描述

一般供應鏈架構包含原物料供應商、製造工廠、倉庫、配送中心、零售商及顧客，其在供應鏈路網上之關係以圖 2.2 表示：

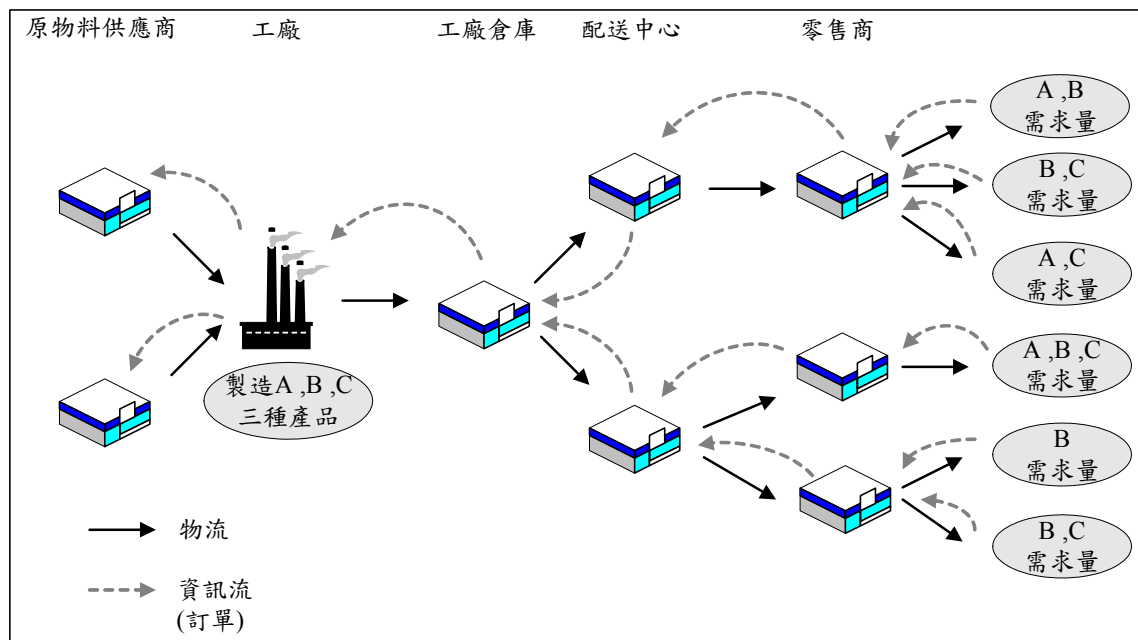


圖 2.2 供應鏈架構圖

資料來源：Perea-Lo'pez et al., 2003

一般而言，供應鏈是依據由零售商所獲得之顧客訂單進行營運，營運過程為在每一營運期限間累積所收到之訂單，如累積全日顧客訂單，並在隔日之前完成訂單需求；若是訂單在營運期限截止後送達，則排到下一階段(隔日)處理。在營運期限到達後，如一天結束後，零售商將會處理累積之訂單，如顧客所需產品有存貨可供應時則將產品包裝並運送給顧客，如無存貨可供應則會產生未結清訂單(back-orders)，甚至是失去該筆訂單。因此，產品的可獲得性是維持顧客滿意度的關鍵因素。

為了能及時滿足顧客需求，零售商必須推估預測未來需求，規劃各相關供應點的訂單排程以獲得產品。然而，若是訂貨量多於顧客需求，將會增加存貨持有成本；反之，如訂貨量不足時，則會造成顧客服務滿意度下降，因此應能準確預測，並適當規劃訂貨量。上述接受



訂單、需求預測、訂單決策過程，會持續往上游進行直到達到製造商。在工廠管理方面，主要是面對原物料取得及生產排程的問題，如果發生原物料取得不及或是生產之產品量少於需求量，則會發生產品短缺的狀況。

因此，如何在最大化系統利潤的目標下進行決策，為一研究課題。在供應鏈路網中所產生之問題簡言之就是，在一供應鏈架構下，在包含各時段的時間平面下滿足顧客需求，而其中應達成之目標如下：

1. 最佳化物料取得規劃。
2. 生產排程。
3. 配送點與配送路網上存貨水準的訂單策略，在最大化利潤下盡可能滿足顧客服務水準。

## 2.3 供應鏈管理的求解方法

而為了求解上述問題，在供應鏈管理上可以使用分散型或是集中型管理法進行求解。分散型方法適用於當供應鏈中的物件分屬於不同公司時，且彼此不會分享內部資訊的情形時；集中型為一整體性方法，協調整體供應鏈中的決策，當供應鏈中全部或大部分的物件屬於同一家公司時，則適用於集中型管理法。以下依分散型與集中型管理法將相關文獻予以分類(Perea-Lo'pez et al., 2003)：

表 2.1 分散型方法

作者(年代)	應用方法	特色	限制
Towill (1982, 1996)	<ul style="list-style-type: none"> <li>轉移函數分析</li> <li>前饋控制</li> </ul>	<ul style="list-style-type: none"> <li>可平滑控制配送路網以及抑制因需求變異所帶來的存貨波動</li> </ul>	製造商的前置時間設定為一固定值
Cetinkaya and Lee (2000)	<ul style="list-style-type: none"> <li>供應商管理庫存</li> </ul>	<ul style="list-style-type: none"> <li>提供一存貨與運輸決策同步的架構</li> <li>適用於配送路網具有大量多樣產品性</li> </ul>	不適用製造商，因模式未考量之
Perea, Grossmann, and Ydstie (2000)	<ul style="list-style-type: none"> <li>混合動態模擬模式</li> <li>標準分散式控制排程</li> </ul>	<ul style="list-style-type: none"> <li>能重現真實系統中的問題，如需求擴大問題或是長鞭效應</li> <li>控制策略在配送路網及生產設施上</li> </ul>	
Sivakumar (2001)	<ul style="list-style-type: none"> <li>離散事件模擬方法</li> <li>多目標線性法</li> <li>自我迴歸模式</li> </ul>	<ul style="list-style-type: none"> <li>著重在製造商設置上並規劃延伸此一方法應用在整個供應鏈上</li> </ul>	無考慮系統中配送路網的緩衝作用及需求單一擾動

表 2.2 集中型方法

作者(年代)	應用方法	特色	限制
Forrester (1961)	<ul style="list-style-type: none"> <li>• 間斷時間動態質能平衡</li> <li>• 非線性延滯</li> </ul>	<ul style="list-style-type: none"> <li>• 最早提出求解動態管理問題</li> <li>• 第一個考量模擬工業系統的品質以及其行為</li> </ul>	由於模式的複雜度和數位電腦的限制，所以只能應用在小型範圍的案例上
Backx, Bosgra, and Marquardt (1998)	<ul style="list-style-type: none"> <li>• 以分解及協調技術求解複雜問題</li> </ul>	<ul style="list-style-type: none"> <li>• 同時求解整合製造商與配送路線，且環境因素不為外部干擾，策略均考慮環境的動態性及其干擾性</li> </ul>	
Shah (1998)	<ul style="list-style-type: none"> <li>• 設定目標法</li> </ul>	<ul style="list-style-type: none"> <li>• 回顧到目前為止整合規劃與排程方面的發展</li> <li>• 考量多點式供應鏈在未來將會是一重要的研究課題</li> </ul>	
Bose and Pekny (2000)	<ul style="list-style-type: none"> <li>• 預測平面的規劃模型</li> <li>• 模式預測控制法</li> </ul>	<ul style="list-style-type: none"> <li>• 構想是利用規劃工具尋找存貨設置點，以確保所需的顧客服務水準，並依 MPC 法追蹤之</li> </ul>	
Ettl, Feigin, Lin, and Yao (2000)	<ul style="list-style-type: none"> <li>• 等候理論</li> <li>• 共軛梯度演算法</li> </ul>	<ul style="list-style-type: none"> <li>• 提出供應路網模式，得到中間以及終端產品的庫存水準，最小化存貨容量並滿足顧客服務水準</li> </ul>	
Sakawa, Nishizaki, and Uemure (2001)	<ul style="list-style-type: none"> <li>• 模糊方法</li> </ul>	<ul style="list-style-type: none"> <li>• 提出一多產品、多點式供應鏈的產品及運輸運問題</li> </ul>	無考量多產出工廠或產品間的交易
Edgar Perea-Lo'pez, B. Erik Ydstie, Ignacio E. Grossmann(2003)	<ul style="list-style-type: none"> <li>• 模式預測控制法</li> <li>• 混合型線性整數規劃模型</li> <li>• 滾動平面法</li> </ul>	<ul style="list-style-type: none"> <li>• 考量系統中物流及資訊流</li> <li>• 動態最佳化架構供應鏈中每一階層及其互動</li> </ul>	

Perea-Lo'pez et al.(2003)研究結論中提出，在供應鏈管理上，集中型管理法比分散型管理法為佳。因為集中型能觀察到系統中所有可能的不一致性以及限制規定，能在資源協調上有較佳的處理方式，且在工廠與配送路網之間亦能作一平衡。而 Seferlis 與 Giannelos (2004)提出一雙層最佳化控制策略，所提出之方法因可適用於系統中的運輸延遲和隨機需求變異，因此比單層控制系統佳；此外，使用回饋控制來維護產品存貨，可在隨機與確定兩種需求變異下處理供應鏈，對於需求或是運輸延遲的擾動，皆能比集中型最佳化控制排程更快速反應。

## 2.4 供應鏈管理之不確定性因素

因供應鏈路網的複雜性以及其身處在快速變動的環境中，而存在著許多不確定性，這些不確定性因素可能來自運送延遲、機器故障、訂單取消等原因，都直接或間接影響著供應鏈的發展與決策，如何掌握並控制此一不確定性成為企業重要的課題之一，因此需先了解哪些是供應鏈管理的不確定性因素。Whybark & Williams(1976)提出造成環境的不確定性可分為需求面及供給面，需求面的不確定性主要會影響生產排程 (Master Production Scheduling，MPS)的預測值與存貨水準，使得實際生產時間及數量產生變化；供給面的不確定性則會使得交貨時間或數量不確定，導致供貨品質不穩定。Lee(1993) 與 Davis (1993)皆將供應鏈體系裡的不確定性因素，分成三類：

1. 需求的不確定因素：需求的不確定性因素會造成存貨的堆積或缺貨，主要是由於顧客訂單之不定期時間與不規則數量，導致需求產生誤差，影響供應商、製造商或是零售商的採購決策。而廠商為了解決此一需求變動通常會增加存貨以因應之，因此在需求變異越大的情形下，存貨成本會相對越高以滿足顧客需求。
2. 製造的不確定因素：因設備故障或是作業程序疏失所產生之不良品機率。主要會造成產品交貨日期延遲、顧客服務水準降低。

3. 供應的不確定因素：原物料之送達時間、可供應數量以及物料之品質均是影響因素，亦會造成產品交貨日期延遲。

如圖 2.3 所示，在顧客需求預測準確度與供應商績效下，從原物料送至製造商製造產品，再將完成產品送至顧客之過程中，這三種不確定性因素的發生均有可能會影響最終產品的交貨日期，這是由於預測顧客需求可能不準確、供應商供給原物料可能延遲加上生產製造亦可能延遲，最後造成產品運送至顧客的交貨日期延遲。

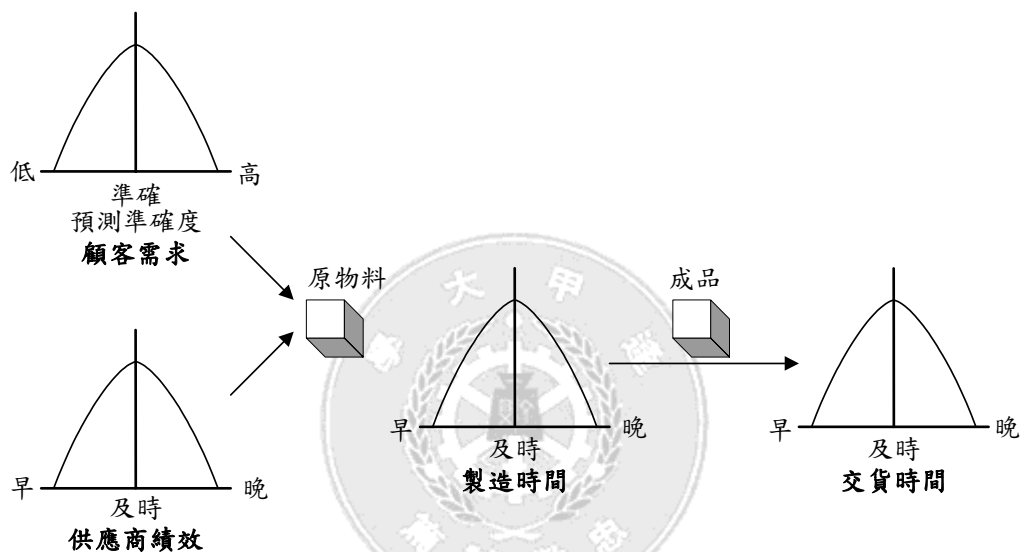


圖2.3 供應鏈管理之不確定性因素的影響

資料來源： Davis, 1993

在不確定因素之因應策略方面，Davis(1993)及 Strader(1998)提出供應鏈三種不確定性中以客戶需求是供應鏈中最主要的不確定性因素，而需求不確定性最主要的原因在於資訊不能正確且即時的傳達，因此要解決需求的不確定性要從預測及資訊分享來著手；Doulgas and James(1993)則認為預測之前必須要有環境的掃瞄以取得相關資訊。(羅國正，2000)將資訊管理策略分為環境掃瞄、預測及資訊分享三方面說明：

1. 環境掃瞄

為避免發生非預期狀況，企業須定時且系統化地不斷掃描企業內

部與外部狀況，以獲得相關資訊。Doulgas and James(1993)提出三種環境掃瞄方式，分別為非正常掃瞄、正常掃瞄及連續性掃瞄：

(1) 非正常掃瞄

為一種反應性偵測。著重在短時間立即反應所掃瞄到之現在到過去對於事件的影響因素，但較少注重在確認及分析未來環境的趨勢和事件的發展。如未來競爭對手合併之變動情形。

(2) 正常掃瞄

為一種預先性偵測。主要是採取規律且系統化地掃瞄企業環境狀況。例如定期調查顧客滿意度、意見或定期評估產品毛利等。

(3) 連續性掃瞄

為一種長期性偵測。不斷地監視環境狀況的發生，企業可以很快地確認、評估及反應環境的變動。例如持續評估搬運的績效。

2. 預測

需求具有容易變動的特性，因此無法真正預期訂單的趨勢狀況，但可藉由預測模式減緩這個問題的嚴重性。Gilbert and Vincent(1981)在研究預測的困難一文提及“沒有比發展具有好的邏輯性及一致性，並能夠反應管理者需求及因應環境變動而快速調整更新的預測系統更為重要的了！”，因此，許多大型企業均花費許多經費在預測上。而使用需求預測的原因有兩個：

(1) 作為運籌管理系統依據

為能減少現有狀況中已存在及潛在的問題，並且能對未來的狀況及挑戰做準備，運籌管理系統需要有一前推式規劃(forward planning)，因此藉由預測結果可進行未來工作之規劃，使得未來的運作能更順暢。

(2) 作為管理決策的依據

經由預測能夠產生出一個未來的架構圖，便能作為管理決策之依據。

目前已發展出相當多的需求預測技術，但必須依據企業特性選擇一個同時具備廣度與深度的預測方法。

### 3. 資訊分享

在資訊快速進步、產品生命週期逐漸縮短且顧客服務水準日趨重要的環境中，上游供應商為了能及時滿足下游廠商對於交貨日期、數量的要求，因此有產生了存貨過量的狀況，造成相當大的存貨成本。且在資訊不足或模糊的影響下，往往因為下游廠商扭曲顧客的需求資訊，造成上游供應商對於顧客需求狀況有著錯誤的判斷。因此加強企業之間資訊分享的技術，已是供應鏈成功的關鍵。透過先進的資訊技術，資訊分享將促使供應鏈的合作夥伴更緊密的協調、合作，以達成整個供應鏈績效的最佳化，資訊分享主要是由各企業人員透過資訊網路之建設，來進行溝通、協調、管理，並適時作出正確之決策，使企業間之合作更加穩固。資訊分享的目標即是能利用現有的資訊技術(如網際網路)，將資訊即時且正確的將有用的資訊往上、下游傳遞給所需的企業，使企業均能獲得正確的需求資訊，以進行存貨、訂單、運送等相關決策，有效減少整個供應鏈的決策誤差，掌握不確定性因素，才能有效降低企業的風險。

## 2.5 時間序列型的預測方法

依分析過程可將預測方法分為定性型(qualitative)與定量型(guantitative)。當歷史資料對預測過程無法提供幫助或是相關影響因素無法量化時，則採用定性型預測方式，主要是利用經驗等主觀判斷法進行預測。

定量型預測法是由觀測值估計得到所需參數，與定性型相較之下顯得客觀許多，時間序列法即為其中一種。所謂時間序列(Time Series)係指一連串以時間順序型態出現之觀測值集合，換言之，是對某動態

系統(Dynamic System)隨時間連續觀察所產生有順序的觀測值之集合。其中包含兩個變數，自變數與應變數，時間為自變數；各時間點所對應的數值資料為應變數。利用時間序列進行預測時有一基本假設，未來的數值資料可藉由過去歷史數值資料進行推估獲得。而在對一時間序列進行推論之前，必須對資料進行分析，尋找其趨勢，產生一假設性的機率模型來代表此一時間序列，如此一來才能進行參數推估、檢視機率模型的適合度以及進行預測。以下介紹幾種時間序列分析法，分別為移動平均法、指數平滑法、迴歸分析法以及自我迴歸移動平均整合法。

### 2.5.1 移動平均法

若資料現在階段的不規則變異可藉由過去之不規則變異加以推估，則此序列便符合移動平均過程；當一時間數列走勢並無呈現趨勢或季節特性時，移動平均法的計算過程可以去除隨機干擾的作用。用移動平均法做預測的優點是計算簡易，對於無明顯的長期趨勢及季節變動的時間數列，預測效果尚佳。缺點則是移動平均之期數  $k$  的決定常無客觀標準，而且用移動平均法對於市場急劇變化的反應也較遲鈍。

因此當需求無顯著趨勢或是季節性時，使用移動平均法 (Moving Average, MA)。這個方法是以過去資料為依據，將最近  $N$  個時段的資料以算術平均或加權平均作為下一時段的預測值。數學式如下，假設當觀測到  $(t+1)$  時段需求後，平均最近  $N$  個時段的資料以作為  $(t+2)$  時段的預測值。

$$L_{t+1} = (D_{t+1} + D_t + \cdots + D_{t-N+2}) / N, \quad F_{t+2} = L_{t+1}$$

$L_t$ ： $t$  時段下的推估水準。

$D_t$ ： $t$  時段下所觀測到的真實需求。

$F_t$ ： $t$  時段下的預測需求。

移動平均法為一簡易的預測法，可消除季節變動，不規則變動所造成之影響。



### 2.5.2 指數平滑法

指數平滑法 (Exponential Smoothing)與移動平均法相似，但卻是用在需求具趨勢或是季節性時使用，唯一不同點在於指數平滑法決定數值時，是利用比重決定資料的重要性差異。數學式如下，本期預測值=前期預測值+(權數)(前期實際值-前期預測值)

$$F_t = F_{t-1} + \alpha(D_{t-1} - F_{t-1}) = \alpha D_{t-1} + (1 - \alpha)F_{t-1}$$

$\alpha$  為平滑係數， $0 < \alpha < 1$ 。當  $\alpha$  接近 0 時，即對過去的觀測值作最大加權。反之，當  $\alpha$  接近 1 時，即對近期的觀測值作最大加權。

指數平滑法之種類很多，上面所介紹的是簡單指數平滑法，其他還有線性指數、應變指數等較複雜的方法。此種方法所需保留的資料少、計算簡易，加上預期的正確性較易獲得，是成本較低且應用廣泛的一種短期預測方法。

### 2.5.3 迴歸分析法

迴歸分析方法基本上是用在線性關係的模型化，為了將迴歸分析應用於非線性關係中，必須先將非線性關係轉換成線性關係，因此當自變數與因變數具有線性相關時，可以快速建立模型，但在非線性問題時則很難建立準確模型。非線性關係包含指數、多項式、對數等。而使用迴歸分析法可能產生誤差變異不均、誤差序列相關、模型過度配適等問題。

### 2.5.4 自我迴歸移動平均整合法

Box 和 Jenkins(1977)所提出自我迴歸移動平均整合法 (Auto Regressive Integrated Moving Average, ARIMA)，簡稱 ARIMA(p, d, q) 應用在需求預測上。ARIMA 包含了三種模型，Autoregressive (AR)、Moving Average (MA) 和 ARMA 三種。主要依據 ARIMA(p, d, q) 三個參數的結果來決定，其中，p 表示自我迴歸過程的階數，d 為差分階數，q 為移動平均過程的階數：當  $d = 0$ ， $q = 0$  時，為 AR(p) 模式； $d = 0$ ， $p = 0$  時，則為 MA(q) 模式。ARIMA 在模式建立上可分

為三個步驟，依序為「模式鑑定」、「模式參數估計」、「模式檢定」，圖 2.4 為 ARIMA 標準流程圖：

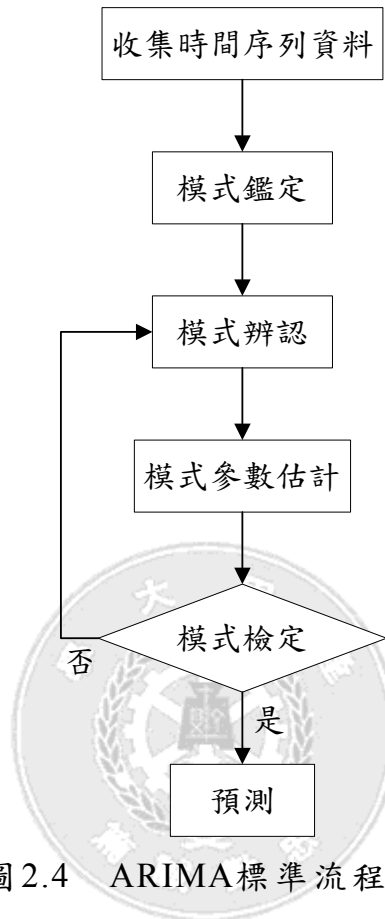


圖 2.4 ARIMA標準流程圖

### 1. 模式鑑定

ARIMA 模式必須建立在平穩的數列資料，然而時間序列的變動不一定都呈現平穩狀態，因此一般的作法是先藉由連續差分 (Differencing) 步驟，將原資料數列化為穩定數列後方可進行 ARIMA 分析，在得到差分的參數  $d$  之後，開始尋找 ARMA ( $p, q$ ) 模型的參數。而參數  $d$ 、 $p$  和  $q$  分別是利用偏自相關變數 (Partial Autocorrelation Function, PACF) 和自相關變數 (Autocorrelation Function, ACF) 觀察而來。如模式中的 ACF 維持許多期的正相關，且 ACF 的值是很緩慢的遞減到 0，顯示其數列資料為一無定向型數列資料，必須對模式進行差分，直到數列轉換成平穩數列，也就是 ACF 快速遞減到 0，而該數列所需差分次數即為  $d$  值。PACF 為幫助分析者判斷該時間數列是否屬於自我迴歸 (Autoregressive Process, AR)，如果 PACF 有被截尾

(Cut Off)的跡象，則為 AR 模式；ACF 則可以幫助鑑定模式數列是否為平穩型或無定向型、數列是否具有移動平均(Moving Average)及數列是否具有季節性因子之特性等，如果 ACF 有被截尾 (Cut Off)的跡象，則為 MA 模式。以下介紹 AR(p)模式、MA(q)模式與整合的 ARMA(p, q)模式之標準數學式 (Box-Jenkins, 1976)：

(1) AR(p)模式

$$\tilde{Z}_t = \phi_1 \tilde{Z}_{t-1} + \phi_2 \tilde{Z}_{t-2} + \dots + \phi_p \tilde{Z}_{t-p} + a_t, \quad \tilde{Z}_t = \tilde{Z}_t - \mu$$

(2) MA(q)模式

$$\tilde{Z}_t = a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \dots - \theta_q a_{t-q}$$

(3) 整合的 ARMA(p, q)模式

$$\tilde{Z}_t = \phi_1 \tilde{Z}_{t-1} + \phi_2 \tilde{Z}_{t-2} + \dots + \phi_p \tilde{Z}_{t-p} + a_t - \theta_1 a_{t-1} - \dots - \theta_q a_{t-q}$$

ARMA 模式中，常使用後移運算因子 (Backward Shift Operator) B 來表示時間延滯。因此定義後移運算因子 B 為

$$B^1 Y_t = Y_{t-1}, \quad B^2 Y_t = Y_{t-2}, \quad B^p Y_t = Y_{t-p}$$

所以利用 B 可改寫為

$$(1 - \phi_1 B^1 - \phi_2 B^2 - \dots - \phi_p B^p) \tilde{Z}_t = (1 - \theta_1 B^1 - \theta_2 B^2 - \dots - \theta_q B^q) a_t$$

$$\text{令 } \phi_p(B) = 1 - \phi_1 B^1 - \phi_2 B^2 - \dots - \phi_p B^p, \quad \theta_q(B) = 1 - \theta_1 B^1 - \theta_2 B^2 - \dots - \theta_q B^q$$

$$\text{則 } \phi_p(B) \tilde{Z}_t = \theta_q(B) a_t$$

為解決非定態數列 (nonstationary)的問題，在 ARMA 模型中加入差分

的動作以是數列能成為定態數列，首先定義差分運算子  $\nabla$  (Difference Operation)，一次差分可表示為

$$\nabla Z_t = Z_t - Z_{t-1} = (1 - B)Z_t$$

假設有  $N$  個觀察值的時間序列，經  $d$  次差分後，成為有  $(N-d)$  個觀察值的時間序列

$$w_t = \nabla^d Z_t$$

可將 ARMA 模型改寫成

$$\phi_q(B) \nabla^d Z_t = \theta_q(B) a_t$$

也就是 ARIMA 模型。

$\tilde{Z}_t$ ：代表  $t$  時點之觀察值。

$\mu$ ：時間數列之平均值，為固定參數值。

$\phi_1$ ：自我迴歸參數，代表權數，為固定的參數值。

$\theta_1$ ：移動平均參數，代表權數，為固定的參數值。

$a_t$ ：誤差項。

$\phi_q(B)$  與  $\theta_q(B)$  分別為  $p$  階與  $q$  階的  $B$  多項式

自相關函式 ACF 為

$$\rho_k = \phi_1 \rho_{k-1} + \phi_2 \rho_{k-2} + \cdots + \phi_p \rho_{k-p} \quad k \geq q+1$$

偏自相關函式 PACF

$$\rho_j = \phi_{k1} \rho_{j-1} + \cdots + \phi_{k(k-1)} \rho_{j-k+1} + \phi_{kk} \rho_{j-k} \quad j = 1, 2, \dots, k$$

$\phi_{kk}$  為 PACF

## 2. 模式參數估計

以最小平方法對未知參數進行推估，未知參數包含  $\mu$ 、 $\phi_1$ 、 $\theta_1$ 。  
若是在樣本數夠大的情形下，可採用最大概似估計法進行推估。

### 3. 模式檢定

以殘差檢定，判斷估計的殘差值序列之分配是否與白噪音假設不符合，若不符合時則必須進行模式的修正。

自我迴歸整合移動平均主要是處理非平穩時間序列資料，可說是迴歸分析的延伸，或是自我迴歸與移動平均的整合，在模型中不但考量變數本身的時間序列影響外，也同時考量了誤差項的時間序列影響。因此本研究在預測方法的選擇上採用自我迴歸整合移動法。

## 2.6 模式預測控制

模式預測控制 (MPC, Model Predictive Control) 為一控制理論，用來求解間斷時間最佳化控制問題，視所設計的控制準則，最大化或最小化目標函式。模式預測控制是將系統程序模式化後，以模式預測出未來時間下之程序回應，再以此回應進行即時性的控制決策。因此模式預測控制是包含一近似系統程序之模式，此一模式可依照程序操作變化，預測未來之程式回應，因此，MPC 在模式和程序愈相似時有愈佳的控制效果。劉權輝(2000)提出預測性控制的基本概念如下：

1. 利用過去時間輸入變化，預測未來時間程序的輸出。
2. 估計雜訊效應，即程序輸出之量測值與預測值之差。
3. 根據雜訊效應，可以改進未來時間程序輸出的預測值。
4. 在閉環路誤差最小目標下，經由最適化計算，而得到在未來時間中，輸入應如何變化，方能使輸出之期望值與預測值的誤差得以最小。

模式預測控制的基本架構表示如圖 2.5 所示(蔡柏澧，2003)，圖中方格是特別用來標示該運算單元為非線性程序，P 表示真實系統程序，M 表示系統模式，C 則是代表控制器。在系統中經常出現多輸入

多輸出之情形，而在這些變數間的交互影響時常不容易掌握，因此藉由操作變數與控制變數配對辨識系統程序過程。操作變數(y)為系統輸入變數，控制變數為系統輸出變數受操作變數及擾動(d)影響。依真實系統(P)建構系統模式(M)，以系統模式預測操作變數對控制變數之影響，必較預測所得之控制變數與真實系統輸出之控制變數做一比較，如其差異小於設定點( $y_i^s$ )，則接受決策變數，否則須調整決策變數。

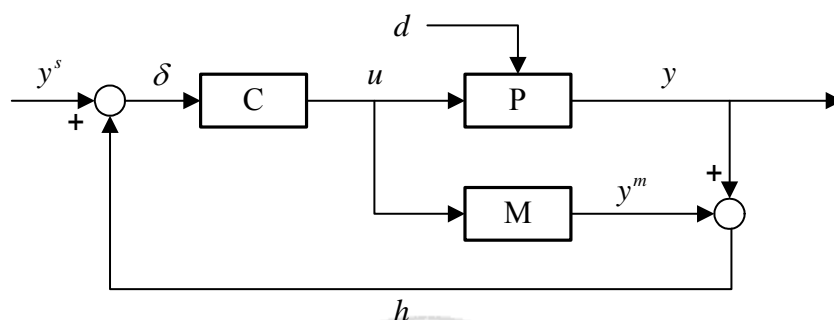


圖 2.5 模式預測控制架構

$d$ ，擾動(disturbances)

$u$ ，操作變數(manipulated variables，MVs)

$y$ ，控制變數(controlled variables，CVs)

可將模式預測控制看成是一個針對系統模式進行最適化操作求解的數學運作。其中，目標式

$$J = \sum_{i=1}^M \sum_{p=1}^p [y_{i,t+p}^s - (y_{i,t+p}^m + h_{i,t})]^2 + \sum_{j=1}^N \sum_{c=1}^C q_{j,c} \Delta u_{j,t+c}^2$$

在滿足所有系統相關的限制之後，藉由目標式透過控制器尋找最佳化的操作變數來控制系統，系統限制式如下：

$$P \geq C$$

$$u_{j,t+c} = u_{j,t+C} \quad (c > C, j = 1, 2, \Lambda, N)$$

$$|\Delta u_{j,t+c}| \leq \Delta u_{j,\max} \quad (j = 1, 2, \Lambda, N ; c = 1, 2, \Lambda, C)$$

$$u_{j,\min} \leq u_{j,t+c} \leq u_{j,\max} \quad (j=1,2,\Lambda,N)$$

$$y_{i,\min} \leq y_{i,t+p}^m \leq y_{i,\max} \quad (i=1,2,\Lambda,M; p=1,2,\Lambda,P)$$

M：控制變數的數目。

N：操作變數的數目。

P：預測時間水平的長度。

C：控制時間水平的長度。

$u_j$ ：第  $j$  個操作變數。

$\Delta u_j$ ：第  $j$  個操作變數之增減量。

$\Delta u_{j,\max}$ ：第  $j$  個操作變數之增減量上限。

$u_{j,\min}$ ：第  $j$  個操作變數之下限。

$u_{j,\max}$ ：第  $j$  個操作變數之上限。

$y_{i,\min}$ ：第  $i$  個控制變數之下限。

$y_{i,\max}$ ：第  $i$  個控制變數之上限。

$y_i^m$ ：由模式M所預測，第  $i$  個控制變數的預測值。

$y_i^s$ ：第  $i$  個控制變數的設定點。

$t$ ：當前取樣控制時間。

$y_i$ ：系統所量測到之第  $i$  個控制變數值。

$q_{j,c}$ ：第  $j$  個操作變數在控制預測時間水平序列中，第  $c$  個取樣控制時刻之限制權重。

$h_{i,t}$ ： $h_{i,t} = y_{i,t} - y_{i,t}^m$ ，第  $i$  個控制變數在當前取樣控制時間，其系統量測值與模式預測值之差。

模式預測控制本身同時包含了前饋控制 (feed-forward control) 和回饋控制 (Feedback control)。前饋控制是利用系統模式向前拓展系統輸出的預測時間水平，針對系統未來的移動軌跡進行預測和控制，達成前饋控制的目的，主要是觀察系統中各種可以測量的輸入量和主要的擾動量，分析它們對系統輸出的影響關係，在這些可測量的輸入量和主要擾動量的不利影響產生之前，透過及時採取糾正措施，以可能的不利影響。回饋控制方面，藉由回傳模式預測值、真實系統輸出值和設定值，在最適化程序中訂定最小化控制誤差目標函數，進而縮減

輸出和設定點之間的誤差。與回饋控制相比，前饋控制克服了回饋控制中因時滯所帶來的問題，改善控制系統的性能。模式預測控制可利用前饋控制，以程序模式與干擾預測為基礎，使系統能預先反應並對擾動做適當反應。模式預測控制依賴兩個重要物件，分別是最佳化工具以及系統模式。最佳化工具作為求解最佳化控制問題，尋找控制策略以最佳化目標函數；系統模式則是計算此一控制策略在系統中未來的回應。

供應鏈管理經導入模式預測控制後，可預測未來需求，有效處理需求變異，依據需求進行最佳化產生決策。同時由於預測控制的多變量模型可以整體性地計算供應鏈中的生產控制條件，並自動地反應運輸延滯及隨機需求變異，調整決策，為供應鏈帶來更多潛在的效益。

## 2.7 滾動平面法

滾動平面法 (Rolling Horizon Approach) 為作業研究中用於預測更新方法之一。Gartner (1982,1983) 建立 OPAC (Optimization Policies for Adaptive Control) 系統，提出滾動平面法方法應用於交通信號控制在動態規劃原理優化控制策略上；而後，Henry 和 Farges (1989) 亦應用滾動平面法於 ProDyn 系統中，ProDyn 為一種綜合的自調式控制系統；Porche (1996) 則將滾動平面法應用於 ALLONS-D 分散型即時交通控制系統上。滾動平面法能在動態環境中藉由相關因素的變化預測未來的行為，隨時間將分析控制期往前滾動，而舊有的預測可藉由新的資料進行修正並向前預測，成功藉由短期較精確的觀測資料進行預測。

圖 2.6 為 RH 觀念示意圖。每一個分析階段 (stage) 含  $h$  個不同時段，稱為投影平面 (Projection Horizon) 或規劃平面 (Planning Horizon)，假設在分析階段  $\sigma-1$  的車流預測後，若於  $1$  時段後可獲得實際資料，則可利用此一短期實際資料配合原有模擬資料，修正分析階段之後半段的預測。



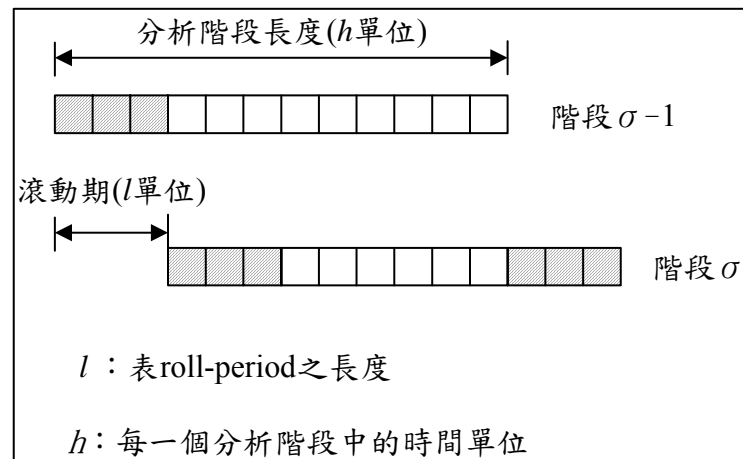


圖2.6 滾動平面法示意圖

資料來源：交通部，2004

以 Gatner (1993) 中 Rolling Horizon 的演算步驟為例：

- 步驟0. 決定投影平面 $h$ 與滾動時段 $l$ 。
- 步驟1. 獲得 $l$ 時段之實際流量資料，並估計 $h$ 時段中的可能流量。
- 步驟2. 根據OSCO (Optimal Sequential Constrained Search Method) 進行最佳化運算。
- 步驟3. 根據步驟2 所獲得的控制，實施 $l$ 時段中的控制。
- 步驟4. 將投影平面向前滾動 $l$ 時段，並建立另一個新的階段 $\sigma$ ，回到步驟1。

本研究中將利用滾動平面法以符合供應鏈中的動態環境。

### 第三章 供應鏈模型架構之建立

本研究主要在利用模式預測控制之架構進行需求預測及控制求得運量決策，以為供應鏈管理帶來最大利潤，因此在研究方法中首先產生供應鏈模式，接著建立模式預測控制以進行需求預測及控制決策。本章將在各節說明相關研究方法，第 3.1 節定義供應鏈問題並構建模式以表達整體供應鏈運作方式，第 3.2 節介紹模擬預測架構，第 3.3 節說明需求預測模式，第 3.4 節構建目標式以進行最佳化。

#### 3.1 供應鏈模式

在建構供應鏈模式之前，需先定義供應鏈問題及其範圍，接著依據問題及範圍構建供應鏈模式。在供應鏈模式中零售商在週期時間點接收到不同顧客、不同產品需求之訂單，而供應鏈的首要目標就是滿足顧客需求，若是顧客需求無法被滿足則會在下一時段產生未結清訂單。為反應供應鏈基礎營運情形，模型中將供應鏈架構視為一路網，以節點代表各階層，依據以下假設進行模式構建：

1. 供應鏈架構中之各階層皆與其上下游有其關連性，非獨立個體。
2. 階層間之相關性皆以方程式表達之。
3. 上游階層對下游階層間之路徑均為有效路徑，均可進行運送貨物的動作。
4. 各階層間之有效路徑皆有運送時間及運送成本之考量。
5. 假設產品與各所需物料之單位比例為1：1。
6. 不考量貨車運量限制。
7. 除零售商外其餘皆無未結清訂單之情形，因此除零售商與顧客間之運量外，其餘各階層間之上游運至下游之運量均代表下游對上游的訂單情形。

### 3.1.1 定義供應鏈問題

研究中所設計之供應鏈形式為一多階層供應鏈，包含工廠、倉庫、配送中心以及零售商，一共包含四個階層。以下為其供應鏈路網圖，顯示階層之間上下游關係，以及有效路徑：

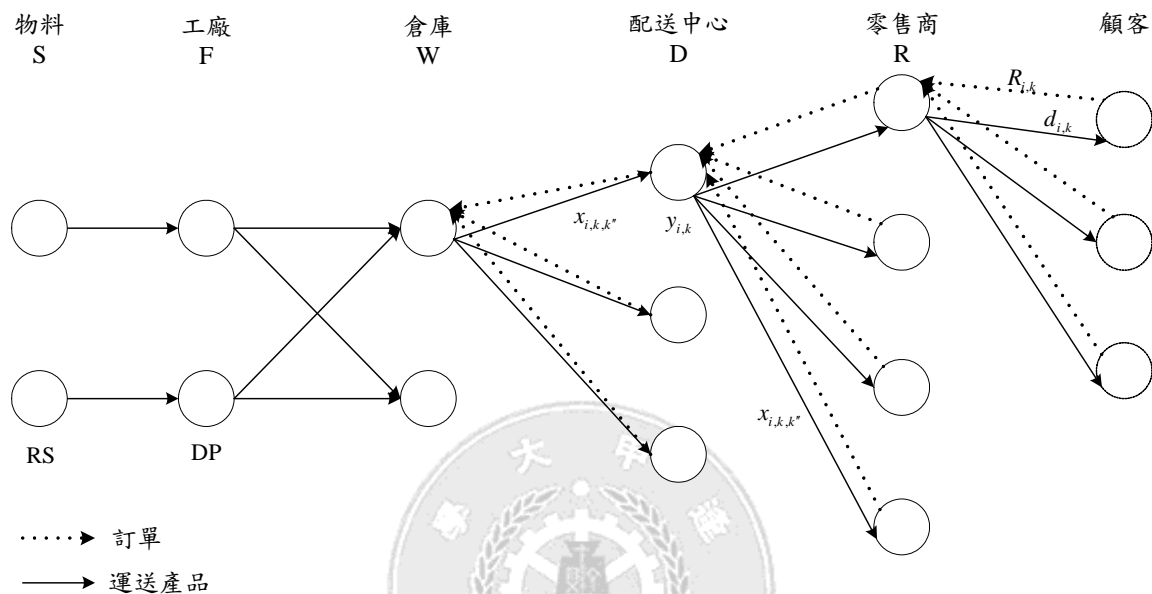


圖3.1 供應鏈路網圖

表 3.1 變數表

S為物料商

D為配送中心

F為工廠

R為零售商

W為倉庫

DP為供應鏈中產品的集合

RS則為物料來源

$y_{i,k}$  :  $i$  產品(物料)在  $k$  節點的存貨量。

$x_{i,k',k}$  : 在上游  $k'$  運送至  $k$  路徑中,  $i$  產品(物料)的總運量。

$x_{i,k,k''}$  : 在  $k$  運送至下游  $k''$  路徑中,  $i$  產品(物料)的總運量。

$g_{i,k}$  :  $i$  產品(物料)在  $k$  節點的產量。

$L_{k',k}$  : 路徑  $(k',k)$  中, 運輸時間。

$P_{i',i}$  : 物料  $i'$  生產成  $i$  產品(物料)所需之時間。

$d_{i,k}$  :  $i$  產品在  $k$  節點的真實運送量。

$R_{i,k}$  :  $i$  產品在  $k$  節點的訂單數。

$BO_{i,k}$  :  $i$  產品在  $k$  節點的未結清訂貨量。

$LO_{i,k}$  : 未能在合理時間內滿足需求所損失之訂單; 損失訂單通常以  $t$  時間下, 以佔未滿足需求之比例顯示。

$VRS_k^{\max}$  : 物料在  $k$  節點的存貨上限。

$VDP_k^{\max}$  : 產品在  $k$  節點的存貨上限。

$T_{k,k''}^{\max}$  : 路徑  $(k, k'')$  之運量上限。

$C_{j,k}^{\max}$  : 資源  $j$  的最大可使用量。

假設在工廠、倉庫、配送中心間的訂單數與運送產品數相同，因此在後續模式構建中將直接以運送商品數代表訂單數；而零售商則有未結清訂單之情形，運送產品數量與訂單數不一定相同，因此零售商對顧客的運送產品數與訂單數會分別討論之。

### 3.1.2 模式構建

依據上一小節所定義之供應鏈問題，可知供應路網的運作流程是將原物料運送至工廠，工廠在生產完產品之後，將產品運送至倉庫庫存，依據顧客需求再運送至配送中心，或是直接運送至零售商。研究中利用一離散時間模型描述路網的動態特性，假設系統會在每一時段(如每日、每週或每月)進行決策，時段的長短則取決於供應鏈路網的動態特性。

供應鏈系統則是由工廠、倉庫、配送中心以及零售商這幾個節點所構成。對每一產品  $i$  而言，在生產時皆有其所需物料標記為  $i'$ 。在每一個節點  $k$  而言，其上游節點標記為  $k'$ ，負責供應節點  $k$ ；下游節點標記為  $k''$ ，可由節點  $k$  供應。所有有效  $(k', k)$ 、 $(k, k'')$  都會在路網中有其可允許路徑。

為完成整體供應鏈架構，呈現供應鏈中各階層間之關係，因此首先構建供應鏈模型，考量各階層本身之存貨平衡、生產限制、各階層間之運量限制以及零售商之未結清訂單，依生產、存貨、運量、未結清訂單三部份構建模式。

以下所構建之模型包含工廠之產量，物料、倉庫、配送中心、零售商之存貨平衡，零售商運送至終端顧客之運量判斷，倉庫、配送中心、零售商之最大可允許存貨水準，供應路網有效路徑的最大可允許運量，工廠之最大生產量，未結清訂單之計算。將其分類為產量、存貨、運量、未結清訂單三部份說明。

#### 1. 產量限制

考量工廠生產產品所需之物料，對於產品產量限制定義為產量需

小於所需之個別物料之上期存貨，與考量物料運送所需之前置時間下的本期物料進貨數之總合。

$$g_{i,k}(t) \leq y_{i',k}(t - P_{i',i} - 1) + \sum_{k'} x_{i',k',k}(t - P_{i',i} - L_{k',k})$$

$$\forall k \in F, t \in T, i \in RS \dots\dots\dots (3.1-1)$$

## 2. 存貨平衡

在存貨平衡方面，考量路網中所有節點皆需進行流量平衡，就像是從上游至該節點的總供應量與該節點至下游的總運送量，而此一平衡動作會牽涉到各時間點之存貨量，對於倉庫、配送中心與零售商的流量平衡數學式如下：

### • 物料

此處是考量物料在運送至工廠後在工廠之物料存貨情形，因此考量物料運送之前置時間，物料在(t-1) 時段之存貨量加上由上游物料商進貨之運量，並減去考量物料生產成產品所需時間下的產品產量，等於物料在t時間之存貨量。

$$y_{i,k}(t) = y_{i,k}(t-1) + \sum_{k'} x_{i,k',k}(t - L_{k',k}) - g_{i,k}(t + P_{i',i})$$

$$\forall k \in F, t \in T, i \in DP \dots\dots\dots (3.1-2)$$

### • 倉庫

考量貨物運送之前置時間，倉庫在(t-1) 時段之存貨量加上由上游工廠進貨之運量，並減去本期運到下游配送中心之運量，等於倉庫在t時間之存貨量。

$$y_{i,k}(t) = y_{i,k}(t-1) + \sum_{k'} x_{i,k',k}(t - L_{k',k}) - \sum_{k''} x_{i,k,k''}(t)$$

$$\forall k \in W, t \in T, i \in DP \dots\dots\dots (3.1-3)$$

• 配送中心

考量貨物運送之前置時間，配送中心在(t-1) 時段之存貨量加上由上游倉庫進貨之運量，並減去本期運到下游零售商之運量，等於配送中心在t時間之存貨量。

$$y_{i,k}(t) = y_{i,k}(t-1) + \sum_{k'} x_{i,k',k}(t-L_{k',k}) - \sum_{k''} x_{i,k,k''}(t) \\ \forall k \in D, t \in T, i \in DP \dots\dots\dots (3.1-4)$$

• 零售商

對零售商而言，存貨平衡是指考量i產品的真實運送量予以修正存貨，因此考量貨物運送之前置時間，零售商在(t-1) 時段之存貨量加上由上游配送中心進貨之運量，並減去本期運到下游客戶之運量，等於零售商在t時間之存貨量。

$$y_{i,k}(t) = y_{i,k}(t-1) + \sum_{k'} x_{i,k',k}(t-L_{k',k}) - d_{i,k}(t) \\ \forall k \in R, t \in T, i \in DP \dots\dots\dots (3.1-5)$$

• 存貨上限

考量工廠、倉庫、配送中心、零售商之存貨容量，為使不同物料及不同產品在同一節點的存貨配置上具有彈性，定義其成品貨物或物料之存貨上限以節點為限，僅分為總物料存貨上限及總產品存貨上限，不分物料總類或產品總類，因此總物料存貨上限如下：

$$\sum_i y_{i',k}(t) \leq VRS_k^{\max} \quad \forall k \in F, t \in T, i' \in RS \dots\dots\dots (3.1-6)$$

總產品存貨上限如下。

$$\sum_i y_{i,k}(t) \leq VDP_{i,k}^{\max} \quad \forall k \in F \cup W \cup D \cup R, t \in T, i \in DP \dots\dots\dots (3.1-7)$$

### 3. 運量

#### • 物料-工廠(資源上限)

將物料供應商此一階層視為一資源提供者，因此僅考量其資源上限，對於製造資源  $j$ ，每一個工廠都有一最大可獲性限制。

$$\sum_{i'} \sum_{k''} x_{i',k,k''}(t) \leq C_{i',k}^{\max} \quad \forall k \in S, t \in T, i' \in RS \quad \dots\dots\dots (3.1-8)$$

#### • 工廠-倉庫

工廠可運至下游倉庫之運量受限於產品存貨量及產品產量，因此所送出之運量必須小於上期產品存貨量加上本期所生產之產量。

$$\sum_{k'} x_{i,k,k''}(t) \leq y_{i,k}(t-1) + g_{i,k}(t) \quad \forall k \in F, t \in T, i \in DP \quad \dots\dots\dots (3.1-9)$$

#### • 倉庫-配送中心

倉庫可運至下游配送中心之運量受限於存貨量及從上游工廠收到之運量，因此所送出之運量必須小於上期存貨量加上本期所收到之貨運量。

$$\sum_{k'} x_{i,k,k''}(t) \leq y_{i,k}(t-1) + \sum_{k''} x_{i,k',k}(t-L_{k',k}) \quad \forall k \in W, t \in T, i \in DP \quad \dots\dots\dots (3.1-10)$$

#### • 配送中心-零售商

配送中心可運至下游零售商之運量受限於存貨量及從上游倉庫收到之運量，因此所送出之運量必須小於上期存貨量加上本期所收到之貨運量。

$$\sum_{k'} x_{i,k,k''}(t) \leq y_{i,k}(t-1) + \sum_{k''} x_{i,k',k}(t-L_{k',k}) \quad \forall k \in D, t \in T, i \in DP \quad \dots\dots\dots (3.1-11)$$

#### • 零售商-顧客

如零售商所接到訂單數小於存貨量時，則能滿足訂單；反之，



則僅能運送出存貨數量，因此針對零售商對顧客之運量首先考量上期未結清訂單，以及因未結清訂單所產生之損失訂單，及當期產生之訂單數。

$$d_{i,k}(t) \leq BO_{i,k}(t-1) - LO_{i,k}(t) + R_{i,k}(t) \quad \forall k \in R, t \in T, i \in DP \dots\dots\dots(3.1-12)$$

除考量訂單情形，所能運至顧客之運量會受限於零售商上期產品存貨及考量貨物運送之前置時間，由上游配送中心進貨之運量。

$$d_{i,k}(t) \leq y_{i,k}(t-1) + \sum_{k'} x_{i,k',k}(t - L_{k',k}) \quad \forall k \in R, t \in T, i \in DP \dots\dots\dots(3.1-13)$$

• 最大可允許運量

在供應路網中每一有效路徑都有其運量上限。

$$\sum_i x_{i,k,k''}(t) \leq T_{k,k''}^{\max} \quad \forall k \in W \cup D \cup R, t \in T \dots\dots\dots(3.1-14)$$

#### 4. 未結清訂單

在各時段下，每一物品無法滿足顧客需求時，紀錄為未結清訂單。本模式僅零售商有未結清訂單產生，因此零售商之當期末結清訂單數等於上一期末結清訂單數加上當期訂單數，減去當期運至顧客之運量已及當期損失訂單數。

$$BO_{i,k}(t) = BO_{i,k}(t-1) + R_{i,k}(t) - d_{i,k}(t) - LO_{i,k}(t) \\ \forall k \in R, t \in T, i \in DP \dots\dots\dots(3.1-15)$$

以上利用數學式建構供應鏈中各階層之間的關係，以及各階層內部所生產產品與物料之間的關係，完成一包含多階層之供應鏈模式。

### 3.2 模式預測控制架構

本研究將模式預測控制應用在供應鏈管理上，主要目標考量：(1)顧客滿意度、(2)供應鏈營運成本，在顧客滿意度方面是藉由未結清訂單達到維持顧客服務水準；營運成本方面則包含存貨成本、生產成本及運輸成本。

模式預測控制 (MPC, Model Predictive Control) 策略架構如圖 3.2 所示，依歷史輸入及輸出值建立系統模式以供預測未來輸出(控制變數)；考量目標函式及限制式進行最佳化得到未來輸入，將未來輸入值藉由系統模式預測輸出，將輸出與設定條件作一比較，如符合設定條件則執行未來輸入；如無法接受則重新最佳化調整未來輸入，以使未來輸出符合設定條件。

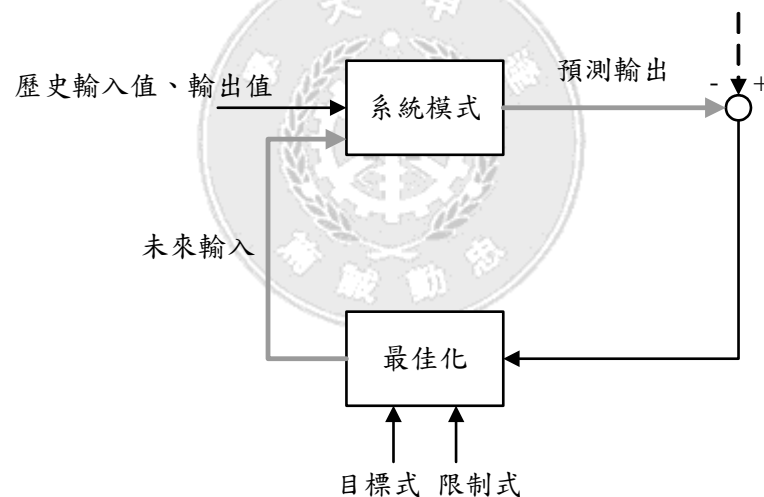


圖 3.2 MPC 策略架構

基於滾動平面法 (rolling horizon approach) 可藉由結合過去及現在的控制行為預測未來控制行為之特性，因此選擇使用此一方法在模式預測控制中進行控制行為預測。在此時間平面下利用建構完成之供應鏈模式預測供應鏈未來行為。在模型中，檢視預測所得之控制變數以調整操作變數，狀態變數如未結清訂單；操作變數如路網上有有效路徑的產品運量與運送至顧客之銷售量。控制行為為最小化依據控制目標所完成之指標，在滾動時間平面  $T_h$  進行計算。在每一個時段中，藉

由比較真實與預測需求值計算出需求擾動與模式不適合之影響，再將此一影響值回饋給模式預測控制。圖 3.3 為模式預測控制概念圖，主要是藉由滾動平面法，定義滾動時間，在每一時段藉由計算出之控制行為於實際情形中執行，所獲得之真實資料以預測下一階段控制行為之反應，依據預測反應更新決策。

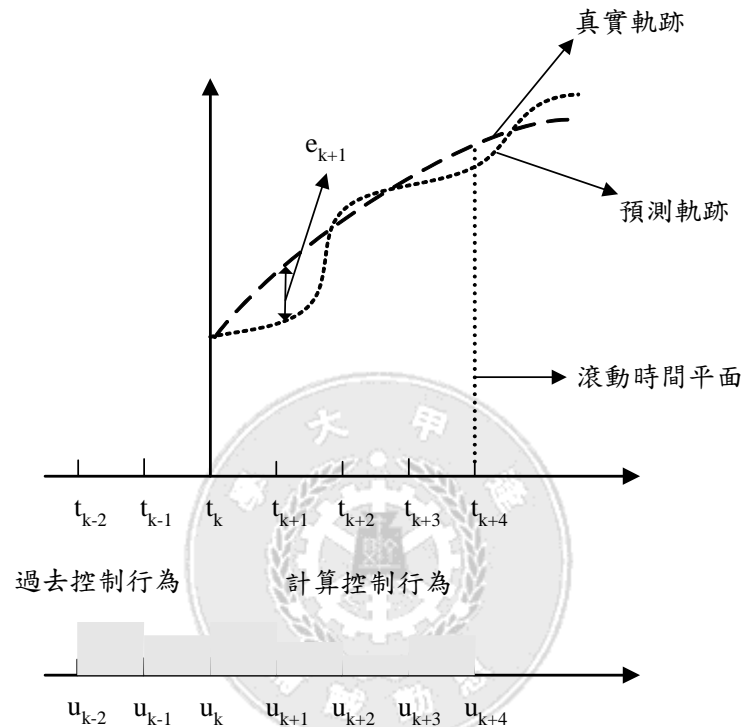


圖 3.3 模式預測控制概念圖

將 MPC 策略架構應用在供應鏈需求預測控制上，即是藉由建構供應鏈模式經由目標式最佳化過程產生未來輸入(如運量決策)，並檢視在供應鏈營運下之輸出值(如未結清訂單)。此外會以滾動平面法向前滾動更新之概念更新需求預測模式與決策變數。

在滾動平面法的概念下，其流程圖如下，定義出分析階段長度( $h$ )及滾動期( $l$ )，以校估出之 ARIMA 模型預測  $h$  長度之訂單需求，模擬供應鏈營運資料以 C 語言寫入目標式及限制式利用 ILOG-CPLEX 進行最佳化求解，獲得未來時段之訂單、運量決策。在獲得決策之後，執行  $l$  時段決策以獲得下一階段之實際訂單需求資料，以往前滾動繼續進行預測及最佳化求解。

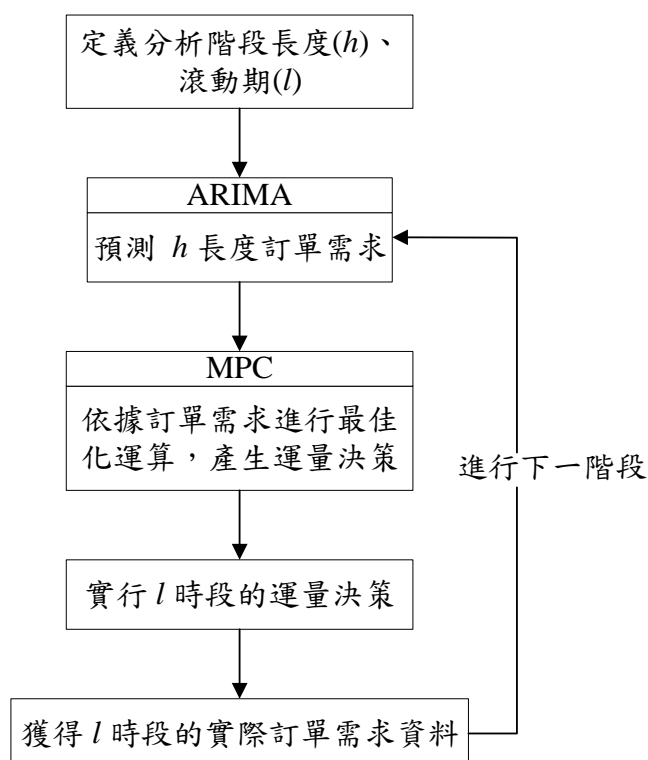


圖 3.4 滾動平面法下決策更新流程

### 3.3 需求預測模式

將 MPC 策略架構應用在供應鏈管理下，將會建立一訂單需求預測模式。依據歷史需求資料校估之模式參數，並進行預測以提供最佳化過程產生決策。訂單需求為一時間序列資料，而時間數列分析是利用過去的資料來判定一個變數的未來行為，以及不同變數在同時段或前後時段之間的關聯性。因此選用時間序列中的自我迴歸移動平均整合法 (Auto Regressive Integrated Moving Average, ARIMA) 以同時考量序列資料之自我迴歸及移動平均特性，建構需求預測模式。藉由 SPSS 統計軟體經歷史需求資料校估出移動平均運算子  $\theta(B)$  及自我迴歸運算子  $\phi(B)$  之參數  $\phi_1, \phi_2, \dots, \phi_p$  及  $\theta_1, \theta_2, \dots, \theta_q$ ，以下為需求預測模式

$$Z(t) = \frac{\theta(B)a(t) + \theta_0}{\phi(B)\nabla^d}$$

$$(1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p)(1 - B)^d Z(t) = \theta_0 + (1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q)a(t)$$

$$Z_t = \theta_0 + \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \dots + \phi_p Z_{t-p} + a_t - \theta_1 a_{t-1} - \dots - \theta_q a_{t-q} \dots\dots\dots(3.3-1)$$

$B$ ：後移運算因子。 $B^1 Z_t = Z_{t-1}$ 。

$\nabla^d$ ：差分運算子。 $\nabla = 1 - B$ ， $d$ 為差分次數。

$\theta_0$ ：常數。

$a(t)$ ：誤差項數列。

$\phi_1, \phi_2, \dots, \phi_p$ ：自我迴歸參數。

$\theta_1, \theta_2, \dots, \theta_p$ ：移動平均參數。

因此以 ARIMA 為預測模式，應用滾動平面法之概念，依不同之滾動期 (roll period) 及分析階段長度 (stage length) 分為數種滾動型態進行預測，實驗中將以不同長度之歷史資料建構模式，將預測數據與實際數據進行比較，選擇適合構建模式之歷史資料長度，以使預測模式切合資料特性，且使預測模型能在動態環境中即時更新。

### 3.4 最佳化模式

在 MPC 策略架構中進行最佳化時需考量目標式、限制式及未來輸出，而當 MPC 應用在供應鏈需求預測控制時，限制式即為供應鏈營運模式，未來輸出則已在供應鏈模式予以考量，目標式則是考量整體效益，也就是供應鏈中的營收及成本。Seferlis 與 Giannelos (2004) 所構建之目標式為最小化存貨成本、運輸成本、未結清訂單、現階段決策變數與前時期之差異；Perea-Lo'pez et al. (2003) 所構建之目標式則為最大化收益減去相關成本，成本方面考量工廠成本、物料由不同反應時間之供應商取得之成本、考量庫存及持有成本之存貨成本。本研究為考量收益及成本兩方面，將產品於零售點之販售收益減去成本，成本包含存貨成本、生產成本、運輸成本以及未結清訂單成本，利用滾動時間平面時間  $T_h$  構建控制指標  $J$ ，數學式如下所示：

Max

$$J = \sum_t^{t+T_h} \sum_{k \in \{R\}} \sum_{i \in DP} \{c_{d,i}(d_{i,k}(t))\} - \sum_t^{t+T_h} \sum_{k \in \{F\}} \sum_{i \in DP} \{c_{g,i,k}(g_{i,k}(t))\} - \sum_t^{t+T_h} \sum_{k \in \{F,W,D,R\}} \sum_{i \in DP \cup RS} \{c_{y,i,k}(y_{i,k}(t))\} \\ - \sum_t^{t+T_h} \sum_{k \in \{F,W,D,R\}} \sum_{i \in DP \cup RS} \{c_{x,i,k',k}(x_{i,k',k}(t))\} - \sum_t^{t+T_h} \sum_{k \in \{R\}} \sum_{i \in DP} \{c_{BO,i,k}(BO_{i,k}(t))\} \quad \dots\dots (3.4-1)$$

$c_{d,i}$  : i 產品販售價格

$c_{g,i,k}$  : i 產品生產成本

$c_{y,i,k}$  : i 產品(物料)在 k 點的單位存貨成本

$c_{x,i,k',k}$  : i 產品(物料)從 k' 運送到 k 的單位運輸成本

$c_{BO,i,k}$  : i 產品在 k 點的單位未結清成本

目標式為最佳化系統效益，其中構建產品利潤、存貨成本、生產成本、運輸成本以及未結清訂單成本，依控制目標完成指標 J 為最佳化原則。

將 3.1 節所構建之供應鏈模式整理如下，為最佳化模式中之限制式：

$$g_{i,k}(t) \leq y_{i',k}(t-1) + \sum_{k'} x_{i',k',k}(t) \quad \forall k \in F, t \in T, i \in RS$$

$$y_{i,k}(t) = y_{i,k}(t-1) + \sum_{k'} x_{i,k',k}(t - L_{k',k}) - g_{i,k}(t + P_{i',i}) \quad \forall k \in F, t \in T, i \in DP$$

$$y_{i,k}(t) = y_{i,k}(t-1) + \sum_{k'} x_{i,k',k}(t - L_{k',k}) - \sum_{k''} x_{i,k,k''}(t) \quad \forall k \in W, t \in T, i \in DP$$

$$y_{i,k}(t) = y_{i,k}(t-1) + \sum_{k'} x_{i,k',k}(t - L_{k',k}) - \sum_{k''} x_{i,k,k''}(t) \quad \forall k \in D, t \in T, i \in DP$$

$$y_{i,k}(t) = y_{i,k}(t-1) + \sum_{k'} x_{i,k',k}(t - L_{k',k}) - d_{i,k}(t) \quad \forall k \in R, t \in T, i \in DP$$

$$\sum_i y_{i',k}(t) \leq VRS_k^{\max} \quad \forall k \in F, t \in T, i' \in RS$$

$$\sum_i y_{i,k}(t) \leq VDP_{i,k}^{\max} \quad \forall k \in F \cup W \cup D \cup R, t \in T, i \in DP$$

$$\sum_{i'} \sum_{k''} x_{i',k,k''}(t) \leq C_{i',k}^{\max} \quad \forall k \in S, t \in T, i' \in RS$$

$$\begin{aligned}
 \sum_{k'} x_{i,k,k''}(t) &\leq y_{i,k}(t-1) + g_{i,k}(t) \quad \forall k \in F, t \in T, i \in DP \\
 \sum_{k'} x_{i,k,k''}(t) &\leq y_{i,k}(t-1) + \sum_{k''} x_{i,k',k}(t-L_{k',k}) \quad \forall k \in W, t \in T, i \in DP \\
 \sum_{k'} x_{i,k',k''}(t) &\leq y_{i,k}(t-1) + \sum_{k''} x_{i,k,k''}(t) \quad \forall k \in D, t \in T, i \in DP \\
 d_{i,k}(t) &\leq BO_{i,k}(t) - LO_{i,k}(t) + R_{i,k}(t) \quad \forall k \in R, t \in T, i \in DP \\
 d_{i,k}(t) &\leq y_{i,k}(t-1) + \sum_{k'} x_{i,k',k}(t-L_{k',k}) \quad \forall k \in R, t \in T, i \in DP \\
 \sum_i x_{i,k,k''}(t) &\leq T_{k,k''}^{\max} \quad \forall k \in W \cup D \cup R, t \in T \\
 BO_{i,k}(t) &= BO_{i,k}(t-1) + R_{i,k}(t) - d_{i,k}(t) - LO_{i,k}(t) \quad \forall k \in R, t \in T, i \in DP
 \end{aligned}$$

本研究所構建之目標式考量了上述限制式中相關的存貨成本、生產成本、運輸成本以及未結清訂單成本，但應用在其他情境時可以依最佳化目的僅考量部份成本再予以最佳化。



## 第四章 模式求解演算法之建立與探討

為求解第三章所建立之模式，本研究利用 SPSS 及 ILOG-CPLEX 進行需求預測模式構建及最佳化求解。本章將在各節探討模式求解演算法之建立方式，第 4.1 節整體模式求解架構介紹整體模式求解過程所使用之軟體，以及求解流程，第 4.2 節需求預測說明構建需求預測模型之流程及選取模型、校估參數之方式，第 4.3 節最佳化求解說明如何利用軟體進行最佳化。

### 4.1 整體模式求解架構

本研究使用統計套裝軟體 SPSS 10 建立 ARIMA 模型。SPSS 是一個組合式統計套裝軟體，具備數據整理、分析過程、結果輸出等功能。SPSS 可採用完全視窗運作管理方式、程式運作管理方式和混合運行管理方式，使用者可根據需求採用不同的運作管理方式。從其統計功能來看，包含了統計類課程的許多方法，例如相關分析、方差分析、聚類分析、迴歸分析等。而本研究則是使用其在數列資料處理之功能校估 ARIMA 之參數，並進行預測。

在最佳化求解工具選擇上採用 ILOG 公司所發展的 CPLEX 數學規劃軟體，在 PC 平台進行求解。CPLEX 在近十年來廣泛的應用在大規模的問題求解上，因它具有程式擴充彈性，可單獨在 DOS、WINDOWS、UNIX 環境下操作求解，亦可搭配如 C、C++、Fortran 及 MPL 程式語言，以撰寫程式呼叫 CPLEX 函式計算，對大規模複雜之問題具有修改上的便利性。此外，CPLEX 對於大規模之線性規劃、混合整數規劃、二次方規劃問題都能在短時間得到模型較優解。因此選用 CPLEX 對不同類型之目標是進行最佳化求解獲得決策。

整體模式求解如流程如圖 4.1，以 SPSS 統計軟體求解 ARIMA 模型以預測訂單需求，將所預測獲得訂單資料代入系統模型中，方程式代入 ILOG-CPLEX 求解線性規劃模型，以決定各點間之運量。



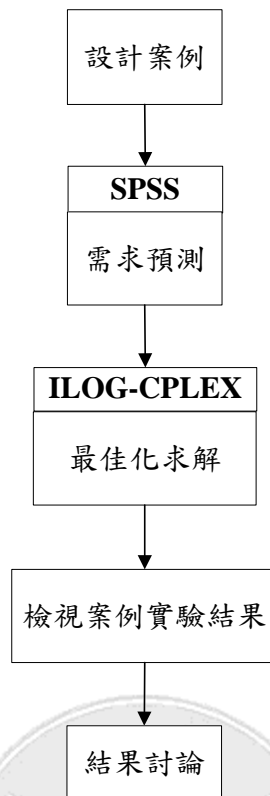


圖 4.1 演算流程圖

## 4.2 需求預測

研究中主要是針對終端客戶對零售商之訂單需求進行預測，因訂單需求資料為時間序列形式，而時間數列的資料特性是每個觀察值之間並非是互相獨立的，因此需利用時間序列分析模式對需求歷史資料進行參數估計已建立預測模型。歷史資料可能包含諸如趨勢、季節、週期等因素，常見的時間序列分析方法主要有：移動平均、迴歸分析、自我迴歸整合移動平均等，研究選用 ARIMA 進行需求預測，因 SPSS 軟體具有 ARIMA 運算功能，而 (陳正昌等，1994)亦提出在 SPSS、SAS、BMDP 三種統計軟體中執行 ARIMA 模型預測的測試上，SPSS、SAS 所產生之模式有較佳的配適性，因此本研究以 SPSS 統計軟體求解 ARIMA 模型，利用歷史訂單數值預測未來訂單數。

需求預測模式產生流程如圖 4.2。研究所使用之 ARIMA 模式是依據平穩的時間序列資料上建立預測模式，首先檢驗數列資料是否為平

穩型數列，穩定型時間數列的定義為若一時間數列  $\{X_t\}$ ，其  $E(X_t)$ 、 $Var(X_t)$ 、 $Cov(X_t, X_{t+s})$  三者必須皆為不受時間  $t$  影響的常數，方可稱  $\{X_t\}$  為一穩定型時間數列，因此依據樣本自我相關函數 (Autocorrelation Function, ACF) 之圖形進行判斷，如 ACF 圖形呈指數型下降速度緩慢 (逐步衰退) 時為非平穩型數列，此時須進行差分直到 ACF 圖形呈快速下降，以使數列資料為平穩型數列，並取得差分階數  $d$ 。接著需判斷  $p$  (自我迴歸階數)、 $q$  (移動平均階數)，在自我迴歸 (Autoregressive Process, AR)、移動平均法 (Moving Average, MA) 及複合型自我迴歸移動平均法 (Autoregressive Process Moving Average, ARMA) 模式中可利用自我相關函數與偏自我相關函數 (Partial Autocorrelation Function, PACF) 確認  $p$ 、 $q$  階數，判別的標準如表 4.1，



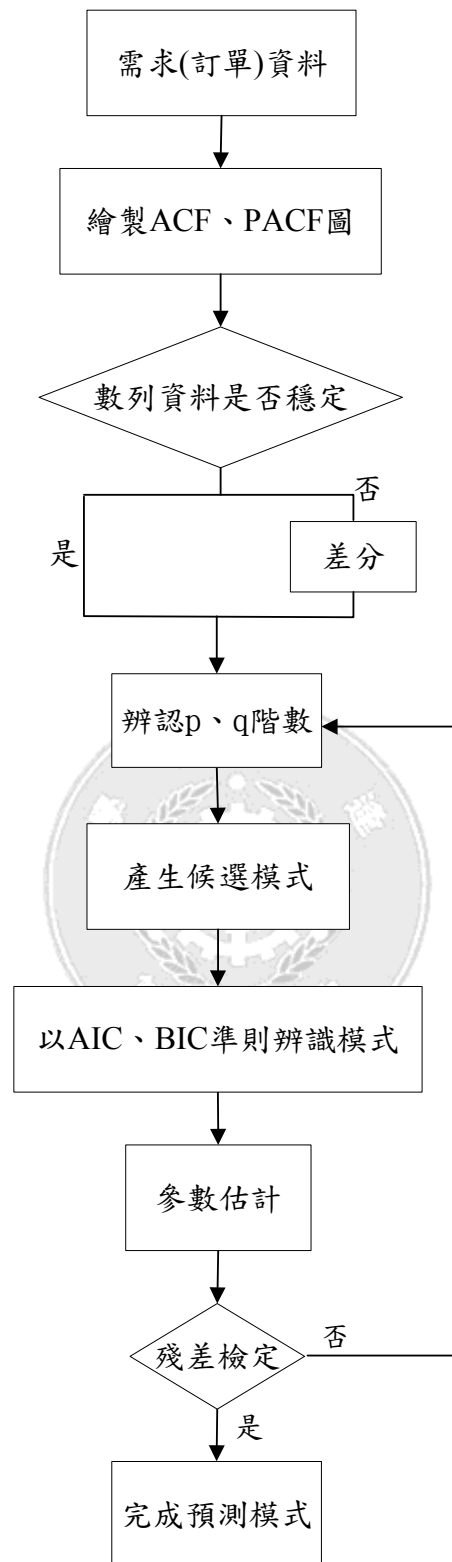


圖 4.2 需求預測模式產生流程圖

表 4.1 AR、MA 及 ARMA 模式判斷標準

	AR(p)	MA(q)	ARMA(p, q)
ACF	拖尾(Tails off)，呈指數遞減或正負相間遞減的形式。	q 期後截尾(Cuts off)。	拖尾(Tails off)，呈指數遞減或正負相間遞減的形式。
PACF	p 期後截尾(Cuts off)。	拖尾(Tails off)，呈指數遞減或正負相間遞減的形式。	拖尾(Tails off)，呈指數遞減或正負相間遞減的形式。

其中，截尾的意義為樣本的 ACF 與 PACF 僅有幾階顯著，其餘階數為 0。在一組時間序列  $\{X_t\}$  中，觀測值  $X_t$  與  $X_{t+k}$  之間的 ACF (在時間上相差 k 期) 為，

$$\rho(X_t, X_{t+k}) = \frac{Cov(X_t, X_{t+k})}{\sigma_{X_t} \sigma_{X_{t+k}}}$$

稱做時間數列  $\{X_t\}$  的自相關函數 (ACF)，以符號  $\rho_k$  表示，差 k 期之 ACF 可寫成

$$\rho_k = \frac{E(X_t X_{t+k}) - \mu^2}{\sigma_x^2}$$

而對任意時差 k，欲將  $X_t$  與  $X_{t+k}$  之間所有變數  $X_{t+1}, X_{t+2}, \dots, X_{t+k-1}$  的影響除去，再考慮  $X_t$  與  $X_{t+k}$  之間的自相關程度時，下列的條件相關係數稱做偏自相關函數 (PACF)，以符號  $\phi_{kk}$  表示之，且稱  $\phi_{kk}$  為相差 k 期的偏自相關函數。

$$\phi(X_t, X_{t+k} | X_{t+1}, X_{t+2}, \dots, X_{t+k-1}) = \frac{Cov(X_t, X_{t+k} | X_{t+1}, X_{t+2}, \dots, X_{t+k-1})}{\sigma(X_t) \sigma(X_{t+k})}$$

由表 4.2 可知：

如果 ACF 圖形僅在  $\phi_{11}$  不為零，且 PACF 圖形呈現拖尾，此時判定時間數列為 MA(1)。

如果 PACF 圖形僅在  $\rho_1$  不為零，且 ACF 圖形呈現拖尾，此時判定時間數列為 AR(1)。

如果 ACF 及 PACF 圖形皆為拖尾情形時，其 ACF 與 PACF 之情形如下表所示，

表 4.2 在 d 次差分下 ACF 及 PACF 之情形

		ACF ( $\rho_k$ )	PACF ( $\phi_{kk}$ )
階數	(1, d, 0)	指數型遞減	僅 $\phi_{11}$ 不為 0
	(0, d, 1)	僅 $\rho_1$ 不為 0	指數型遞減
	(2, d, 0)	混合指數遞減或正負相間遞減的形式。	僅 $\phi_{11}$ 、 $\phi_{22}$ 不為 0
	(0, d, 2)	僅 $\rho_1$ 、 $\rho_2$ 不為 0	混合指數遞減或正負相間遞減的形式。
	(1, d, 1)	從第一階後呈指數遞減	從第一階後呈指數遞減

資料來源：Box-Jenkins，1976

雖然已可將 ACF 及 PACF 圖形應用在模式階數辨認上，但只能產生初步候選階數，仍無法明確決定其階數，因此利用模式選取準則 (Model selection criterion, MSC)，以能在多個候選模式中，選出最佳模式，有效率的考慮模式之精簡性及避免過度高估參數的個數。較常使用二種準則有 AIC 準則 (Akaike's information criterion)、BIC (Schwartz's Bayesian Criterion) 準則。AIC 準則是由 Akaike(1974) 提出，BIC 準則則是由 Schwartz(1978) 提出。其一般模式分別如下：

$$AIC_{p,q} = \frac{-2 \ln(\text{maximized likelihood}) + 2r}{n} \approx \ln(\hat{\sigma}_a^2) + r \frac{2}{n} + \text{constant}$$

$$BIC_{p,q} = \ln(\hat{\sigma}_a^2) + r \frac{\ln(n)}{n}$$

$\hat{\sigma}_a^2$ ：殘差方差。

$$\hat{\sigma}_\varepsilon^2 = \frac{1}{n - (p + q + 1)} \cdot \sum_{t=1}^n (x_t - \hat{x}_t)^2。$$

$x_t$  和  $\hat{x}_t$  表示觀測和估計值。

$r = p + q + 1$ , 為模式中需估計之參數個數。

上式之第一項乃測度模式配適程度，第二項則是對其餘參數之懲罰，因此，當 AIC 準則或 BIC 準則最小值所在階數為最適階數。

在確認(p, d, q) 後以最大概似估計法推估模式參數，接著以 t 值檢定參數 95%之顯著性，當 t 值大於 1.96 時，表示該參數為顯著，方可接受該模式。ARIMA 模型時所建立的最基本假設，也就是誤差項數列必須符合常態且互相獨立的假設，也就是假設為”白噪音”(white noise) 數列，因此需進行殘差檢定，確認模式之誤差項為常態分配，符合白噪音，若不符合則須對模式進行修正。殘差值之檢定如下：

1. 判斷殘差值的 ACF 圖與 PACF 圖。若 ACF 與 PACF 之值均不顯著，則可認定殘差值為白噪音過程。
2. 在統計模式合適之假設條件下，以 Ljung and Box (1978)所提出修正後的 Q 統計量(portmanteau Q-Test)檢定模式。

$$\tilde{Q} = n(n+2) \sum_{k=1}^K (n-k)^{-1} r_k^2(\hat{a})$$

Q 為一統計量近似卡方分配  $\chi^2(K-p-q)$ ； $n = N-d$  為實際殘差之個數；k 為殘差自我相關係數之個數。若  $\tilde{Q}$  檢定值小於  $\chi^2_{\alpha}(K-p-q)$ ， $\alpha=0.05$ ，則表示殘差間無顯著相關，可接受該模式。

### 4.3 最佳化求解

依據第三章所架構出之供應鏈模型及目標式，以 C 語言撰寫程式，並呼叫 CPLEX 函式進行最佳化求解，圖 4.3 為最佳化流程，說明如下：

1. 設定陣列大小

包含限制式行數、列數、非零係數個數。

2. 讀入陣列數值

需先定義目標式係數陣列、限制式非零係數位置陣列、限制式非零係數累積個數陣列、限制式非零係數數值陣列、限制式常數項數值陣列、變數型態陣列、限制式型態陣列大小。

3. 設定問題型態

設定目標式的型態為最大化或是最小化。

4. 模式建立

在所有目標式及限制式設定完成後即可呼叫 CPLEX 函數讀入所有陣列，建立完成模式。

5. 最佳化求解

CPLEX 具備多種解題函數，包含線性規劃、混合型整數規劃、二次方規劃、網路問題，可依前面所建立之模式形態予以呼叫使用。

6. 輸出模式及最佳值

CPLEX 可輸出完整模式及最佳解，以供模式校正。

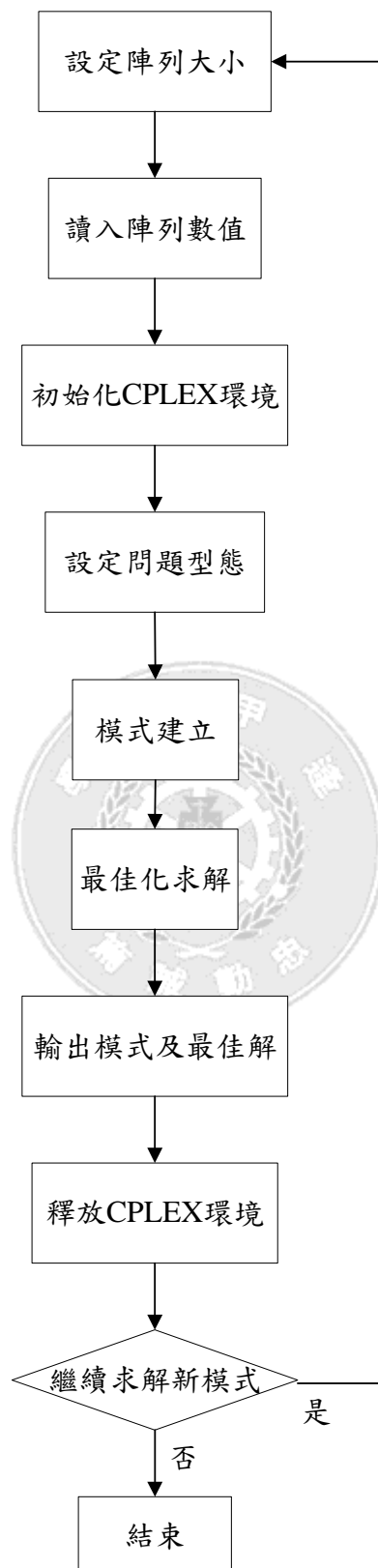


圖 4.3 CPLEX最佳化求解流程



## 第五章 數值實驗

為構建模式及驗證模式之適用性，本章將以自行車產業為基礎設計案例架構，以自行車每月出口至各外銷國家之數據資料搭配滾動平面法之概念校估出需求模式之參數，並以虛擬之供應鏈資料進行模式預測控制模型之數值實驗，以了解本研究所建構之模式的特性。第 5.1 節介紹自行車產業背景，說明需求模型及最化決策之實驗設計，列出實驗所需之數據資料；第 5.2 節進行多階層單一產品實驗，依據資料建立需求預測模型及進行最佳化產生決策，以敏感度分析檢視模式決策結果；第 5.3 節以自行車產業為基礎設計的多階層多產品案例進行實驗，依據資料建立需求預測模型及進行最佳化產生決策，以敏感度分析檢視模式決策結果。

### 5.1 實驗設計

在數值實驗部份是以實際產業為案例設計實驗，首先介紹該產業之發展與營運概況，接著依據該產業之特性進行案例設計，並藉由實際與虛擬數據資料建構模式進行數據實驗。以下透過案例介紹、案例設計與實驗資料分別說明。

#### 5.1.1 背景介紹

本研究將供應鏈產業著眼於自行車產業，自行車產業發展的歷史從一開始台灣無自行車行業，全需仰賴日本進口，到創立本土自行車品牌，一度達到全台自行車的黃金時期，然而在 1968 年日本輸入 50c.c. 機車，自行車需求量大幅下降，此時自行車產業轉而朝向外銷市場，並再度湧現訂單，在 80 年代時出口量及出口值皆達到巔峰，但在 2000 年面對大陸較低人工成本的優勢，使得自行車出口量大幅衰退，因此本土自行車產業進入「轉型期」，著重於加強產品研發，強化與大陸產品的差異性。而現在的自行車產業具有上中下游體系緊密結合，以及為配合環境與市場佈局全球的特性，因此選擇自行車產業作為實驗案例。

由2000年進入轉型期到今年2005年自行車產業在出口方面的銷售概況可由圖5.1~5.3所示，出口量2000年曾高達80萬台，在2004年乙減少至30萬台左右，雖然出口量出現減少的趨勢，但因自行車產業朝技術層面發展開發高價位產品，因此出口單價已由最初的100美金提升至200美金，銷售額並非隨出口量減少而遞減。在這期間，自行車產業於2003年成立一非營利組織「A Team」，初期是由巨大機械及美利達工業結合11家零件廠共同組合，至2004年又新增7家會員廠，使A Team的總會員廠數達到20家。此組織主要目標除改善整體產業供應鏈的營運績效，主要是發揮更大的團體力量提昇創意研發能量，有助於台灣自行車的轉型與創新。在改善整體產業供應鏈營運績效上，A Team導入「豐田生產方式(TPS)」以提升產能，使產業具備少量多樣的生產彈性；TPS生產觀念分為兩大概念：及時化(Just In Time, JIT)及自働化。及時化之觀念為「必要的產品，於必要的時間，生產必要的數量」，也就是以最小之物料、半成品、成品和最短之前置時間，達到精確的產量。自働化目的在於防止機器或生產線生產出不良品，與一般自動化之差別在於，自働化是在單純機器的自動化上加上了人工的智慧，也就是當機器在有異常狀況發生時會自動停止，再由人員進行處理，如此一來可增加人員所能負責之機器數量、減省人力，並減低產品不良率。

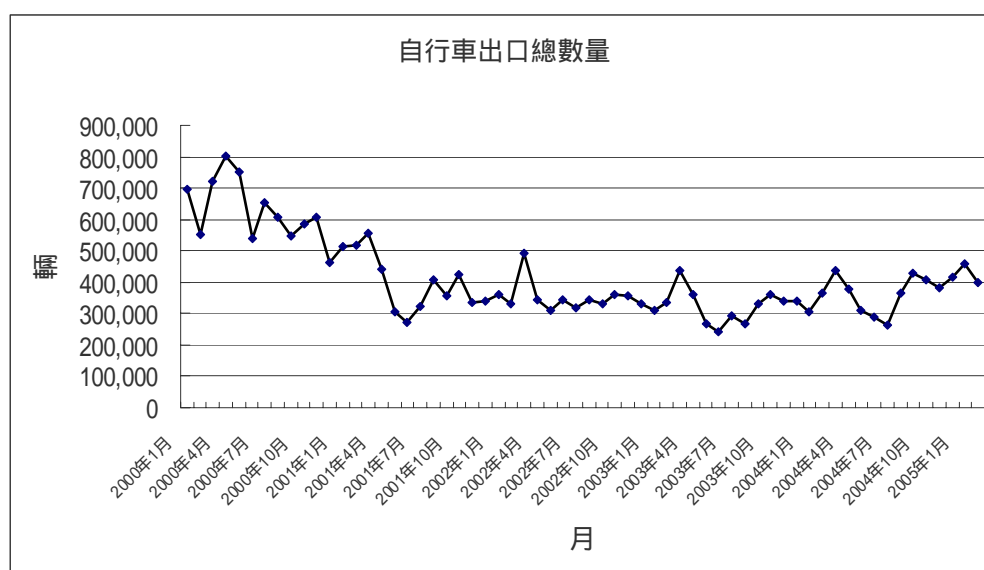


圖 5.1 自行車每月出口總數量

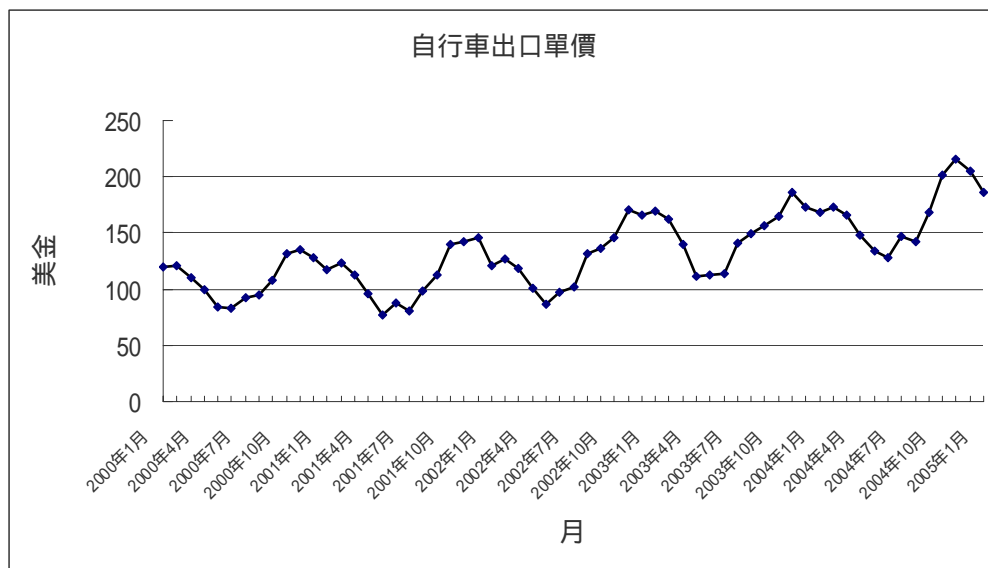


圖 5.2 自行車每月出口單價



圖 5.3 自行車每月出口銷售額

### 5.1.2 實驗設計

在實驗案例設計上，首先設計將供應鏈管理應用在自行車產業上之案例架構，接著建立供應鏈架構中所需之需求預測模式，利用不同形式之滾動方式進行訂單需求預測模型之建立，以能將需求預測模型所預測得到之需求資料載入模式中進行最佳化決策。

## 1. 案例架構

自行車產業結構依產品主要可分為零件及成車兩大種類，其大小零件高達2000個之多，就其功能特性不同可分成六大系統，如圖5.4所示依序為車架系統、傳動系統、車輪系統、轉向系統、煞車系統及附件，在零件完成主要系統組裝後，交由車廠進行車架裝配及車輪裝配，最後進入裝配而完成成車生產。因此可將產業分為上、中、下游，依序為物料供應商、零件商及裝配廠，其間關係非常密切，一旦物料供應商或零件商無法及時滿足下游需求即會導致經銷商無法出貨。

上游		中游		下游	
合金鋼材	中鋼	車架	太平洋、野寶、錦祥、一心、九川、卜威、久敏、巨隆、永久、全順、旭生、韓氏、龍德	成車	巨大、美利達、愛地雅、太平洋、德高、昆誠、崑崙、菲力、競通閣、永輪、永源、中廣、宇駒、明旬行、野寶、伯佳德源
鋁合金		曲柄	天心、嘉德、鑫悅、銳輪、金日、龍銳		
其他合金		花鼓	川飛、鑫元、久裕、達康、崑崙、林三策		
油漆塗料		鏈條	桂盟、大亞、全仕通、雅那		
橡膠	遠大、正新	飛輪	川飛、日馳、久裕、利達、龍德		
		煞車器	利奇、彭星、永傑、騰輝		
		變速器	日馳、川飛、友隆、久裕、榮輪		
		手把	利奇、嘉弘、利隆		

圖 5.4 國內自行車產業上、中、下游關聯圖

資料來源：IEK-IT IS，元大京華投顧製圖，2003/05

因實驗所採用之需求資料為出口資料，因此將終端客戶對象設定為國外廠商；而依據自行車產業上中下游之間關連性極高之特性，案例架構設計如圖5.5。供應鏈上游之原物料可同時供應不同零件商，而零件商可依不同成本及需要不同前置時間區分其特型，提供下游不同性質之裝配廠所需零件；不同裝配廠因其裝配產品不同所需零件亦不相同，因此需由不同零件商負責供應，裝配廠在裝配完成產品後運至經銷商，此案例中之經銷商則為單一經銷商銷售兩種產品，負責依國外廠商不同需求出口產品。

因近年來電動自行車隨環保意識及休閒風潮在自行車產業成長迅速，因此在產品及零件設計上，將產品1視為一般自行車，產品2視為電動自行車；產品所需零件分為低成本及高成本零件，兩種產品所

需之低成本零件其成本相同，但在高成本零件上電動自行車所需之零件其成本則較高。

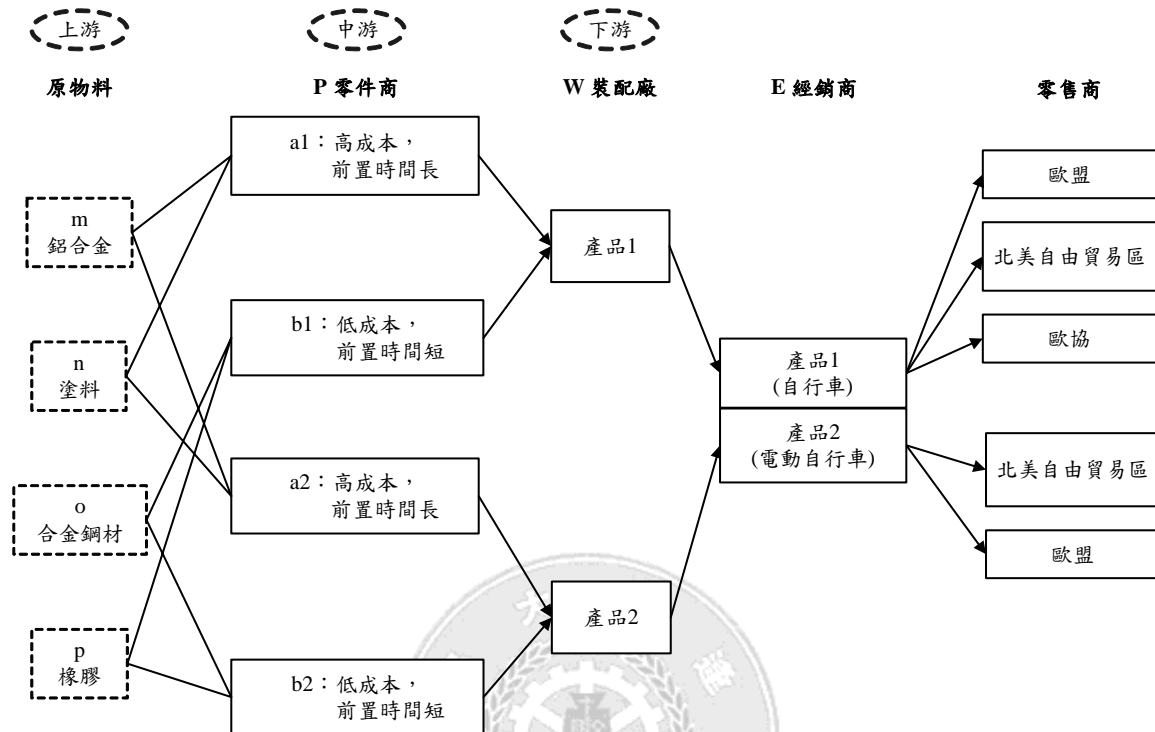


圖 5.5 實驗案例架構

## 2. 需求預測模型

因需求資料都有其特性，如週期特性，或是過早之歷史資料因當年環境與現今環境已大不相同，反而不適用於推估未來需求，因此依不同筆數之歷史資料分成不同型態，以尋找出最適合用來建立預測模型之資料筆數。預測週期的定義應依資料性質予以決定，因實驗數據為自行車出口資料，因此以一年(12個月)資料為基礎，設計不同預測週期。如下圖所示，預測未來12筆資料(為一年，12個月)，分為四種型式：

- A. 以 12 筆歷史資料預測未來 12 筆需求，
- B. 以 24 筆歷史資料預測未來 12 筆需求，
- C. 以 36 筆歷史資料預測未來 12 筆需求，
- D. 以 48 筆歷史資料預測未來 12 筆需求，

實驗中在校估出最佳週期後進行需求預測，取前6筆需求進行最佳化決策，再向前經過6個需求時段後可再更新預測模式進行下一週期預測。

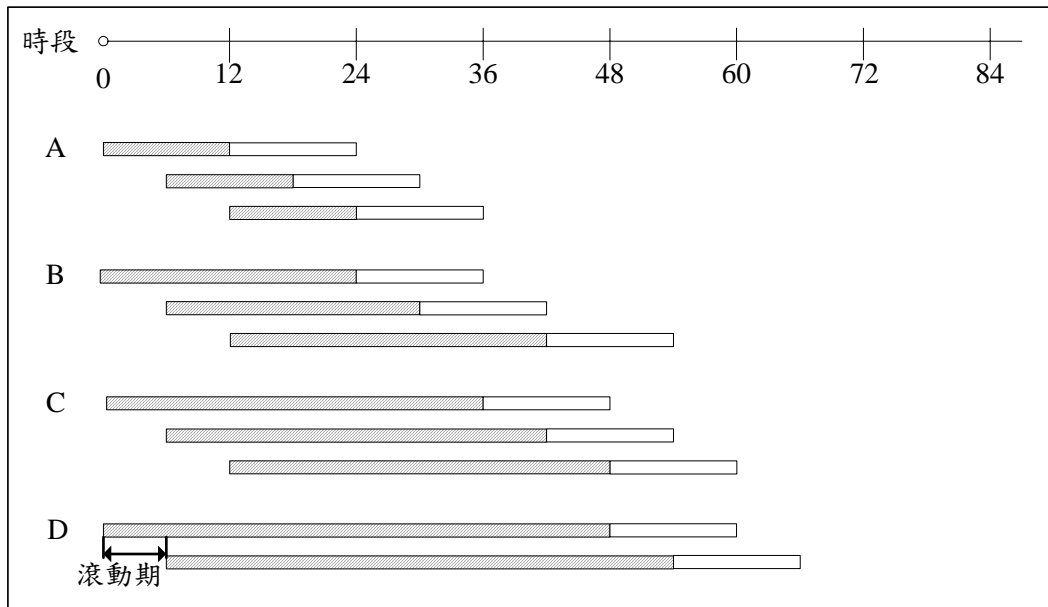


圖 5.6 不同歷史資料筆數之需求預測案例

### 5.1.3 實驗資料

為完成需求預測模型及驗證供應鏈模式，本研究將以一基礎供應鏈之營運資料作為實驗案例，其中包含以下數值資料，在實際資料方面包含訂單需求數主要作為建立需求預測模型之用，此外產品銷售額則是使用在最佳化目標式的銷售收益上；虛擬資料包含存貨量、存貨上限、有效路徑運量上限、運量、未結清訂單數、損失訂貨數、運輸成本、存貨成本、未結清訂單懲罰值，均是在進行最佳化決策時所需之數據資料。

表 5.1 供應鏈基礎資料

資料來源	資料類別	工廠		倉庫		配送中心		零售商		顧客
實際資料	訂單數									✓
	產品銷售額							✓		
虛擬資料	存貨量			✓		✓		✓		
	存貨上限			✓		✓		✓		
	有效路徑運量上限		✓		✓		✓			
	運量		✓		✓		✓		✓	
	未結清訂單數							✓		
	損失訂貨數							✓		
	運輸成本		✓		✓		✓			
	存貨成本			✓		✓		✓		
	未結清訂單成本							✓		

以下首先介紹實際之訂單需求資料及產品銷售額，接著是最佳化決策所需之虛擬資料。

### 1. 實際數據資料

實際數據資料包含訂單需求資料及產品銷售額，來源為「台灣區自行車輸出同業公會」，由該公會網頁(<http://www.tbea.org/index.htm>)上獲得自行車出口數量之資料，可獲得之資料為1998年1月至2005年2月共74筆自行車、電動自行車出口至各國之每月出口量、每月出口單價以及每月銷售額之資料。依實驗設計所需在一般自行車出口國中選取前三大出口國之數據資料，電動自行車出口國中則選取二大出口國之數據資料，數據資料參見附錄一。

### 2. 虛擬數據資料

在最佳化過程中所需之供應鏈營運資料是由不同型態之虛擬資料進行實驗。分別為物料上限、初期存貨量、存貨上限、前置時間、生產(裝配)時間，其中物料上限、及存貨上限設為足夠供應所需要的數量，初期存貨量則設為0。一般物料及零件前置時間為1個月，某些

物料及零件所需之時間較長則為2個月；產品運送至顧客之前置時間皆為1個月。零件及產品的生產時間則均設計成1個月。階層間之前置時間與生產時間如表5.2至表5.6所示：

表 5.2 前置時間(物料-零件商)

物料運送至零件商所需時間(月)		零件商			
		P1	P2	P3	P4
物料	m	2	-	2	-
	n	1	-	1	-
	o	-	2	-	2
	p	-	1	-	1

表 5.3 前置時間(零件-裝配廠)

零件運送至裝配廠所需時間(月)		裝配廠	
		W1	W2
零件商	P1	2	-
	P2	1	-
	P3	-	2
	P4	-	1

表 5.4 前置時間(產品-經銷商)

產品運送至經銷商所需時間(月)		
		經銷商
		W1
裝配廠	W1	1
	W4	1

表 5.5 生產零件時間

生產零件所需時間		零件(月)			
		a1	b1	a2	b2
物料	m/n	1	-	1	-
	o/p	-	1	-	1



表 5.6 生產產品時間

生產產品所需時間(月)		產品	
		產品 1	產品 2
零件	a1	1	-
	b1		
	a2	-	1
	b2		

## 5.2 多階層單一產品實驗

在多階層單一產品的設計下，建立工廠產品運送至倉庫進行存貨，再由倉庫送至配送中心，再由配送中心配送貨物至零售商，將產品銷售給顧客。如圖5.7所示，在倉庫、配送中心、零售商及顧客之間是由訂單數決定運送產品量，而零售商會出現未結清訂單之狀況，因此運送產品量不一定等於訂單數，其於階層間之運送產品量即等於訂單數。以下先建立需求預測模式，再利用預測獲得之需求量進行最佳化獲得決策。

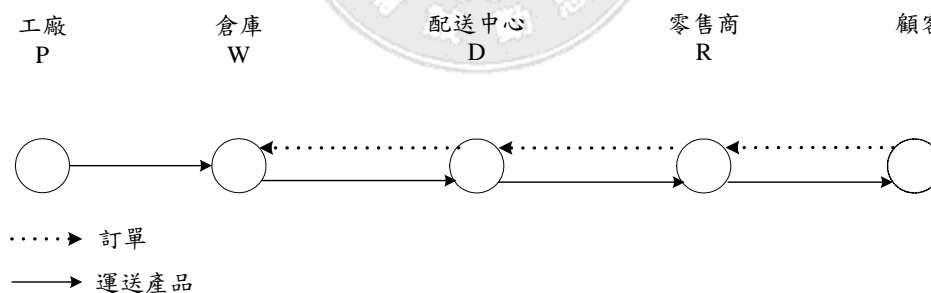


圖 5.7 多階層單一產品供應鏈路網

### 5.2.1 需求預測模型

為提供最佳化所需之需求資料，本節將經由實際數列資料在SPSS中建立需求預測模型。首先以2000年1月~2004年2月共50筆每月自行車出口總數量為實驗數據，將預測結果與2004年3月~2005年2月驗證模式之可行性。判斷需求資料是否為平穩數列，繪製數列資料之ACF圖，由ACF圖形呈指數型下降速度緩慢之情形，可判斷數列為非平穩

型數列，因此對數列進行一次差分，並繪製數列經一次差分後之ACF圖，發現經一階差分後，ACF圖呈快速下降，成平穩型數列，因此差分次數 $d=1$ 。

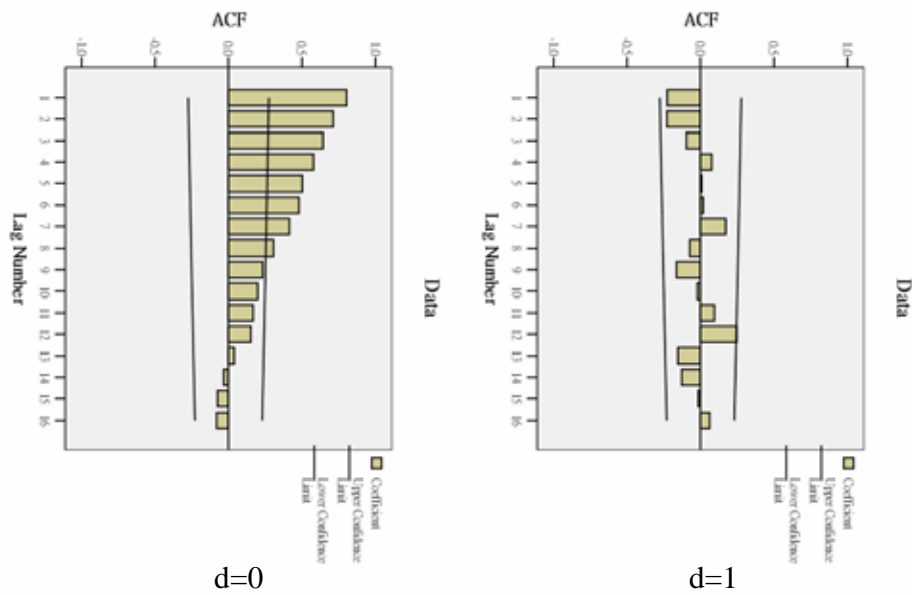


圖 5.8  $d=0$ 、 $1$  下數列資料之 ACF 圖

由圖5.9產生數組候選模式，並將模式分為是否具有常數項，以AIC及BIC準則選取模式，候選模式與其AIC及BIC值如表5.7所示：

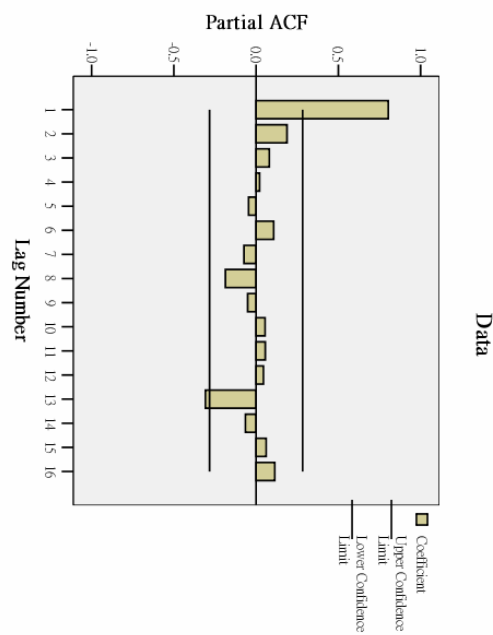


圖 5.9 數列資料之 PACF 圖

表 5.7 候選模式之 AIC 及 BIC 值

(p,d,q)	(1,1,0)	<b>* (1,1,1)</b>	(1,1,2)	(2,1,1)	(2,1,2)
AIC	1244.573	1239.589	1240.806	1239.929	1241.042
BIC	1248.357	1245.264	1248.373	1247.496	1250.501

經由結果發現模式在 $(p,d,q) = (1,1,1)$ 情形下，其AIC及BIC值均較低，因此對該模式進行殘差檢定，首先繪製模型殘差之ACF及PACF圖如圖5.10，發現其ACF及PACF值均不顯著，因此認定其殘差值為白噪音過程，選取 $(p, d, q) = (1, 1, 1)$ 模式。

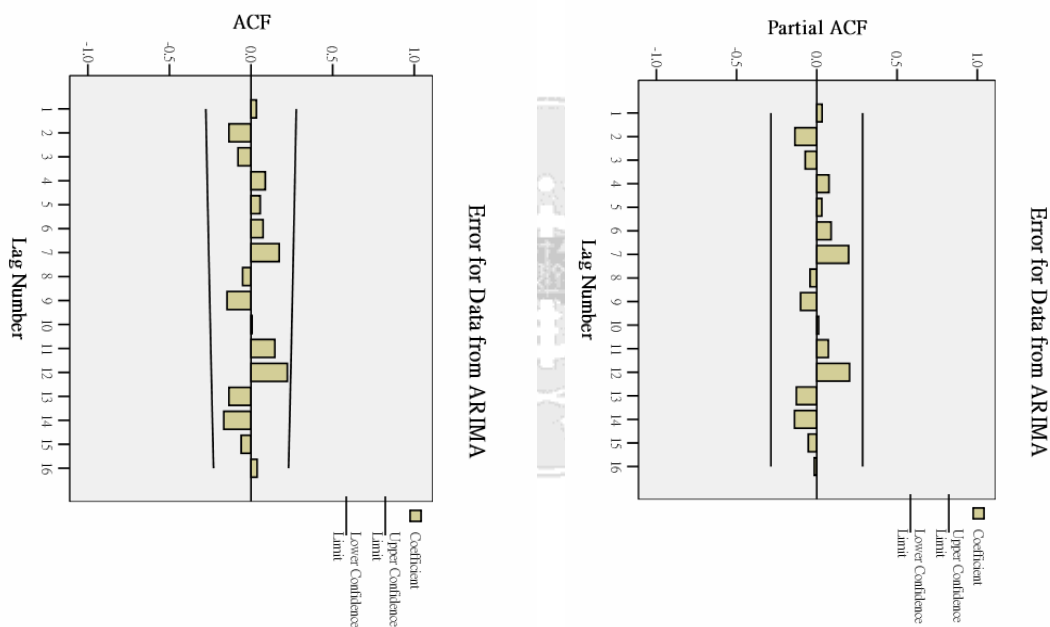


圖 5.10  $(p, d, q) = (1, 1, 1)$  殘差之 ACF 圖、PACF 圖

最後計算其Q統計量，共有16筆殘差自我相關係數值如下：

表 5.8 殘差自我相關係數

	殘差自我相關係數		殘差自我相關係數
1	-0.003	9	-0.090
2	0.040	10	-0.021
3	-0.056	11	0.104
4	-0.041	12	0.229
5	0.071	13	-0.109
6	0.017	14	-0.151
7	0.090	15	-0.138
8	-0.058	16	-0.078

Q統計量方程式如下：

$$\tilde{Q} = n(n+2) \sum_{k=1}^K (n-k)^{-1} r_k^2(\hat{a})$$

N：數列資料個數

d：差分次數

n：實際殘差個數=N-d

k：殘差自我相關係數之個數

因此，

$$n=N-d=50-1=49, k=16$$

$$\tilde{Q} = 49 * (49 + 2) * \frac{0.15}{(49 - 16)} = 16.49$$

若  $\tilde{Q} < \chi^2_{\alpha}(K-p-q)$ ,  $\alpha=0.05$ ，表示殘差間無顯著相關，可接受該模式，

$$\text{而 } \chi^2_{0.05}(16-1-1) = \chi^2_{0.05}(14) = 23.7 > 16.49$$

結果顯示殘差間無顯著相關，可接受該模式，因此預測模式為(p,

$d, q)=(2, 1, 1)$ ，得到預測模式：

$$R(t) = \frac{(1 - 0.76B) * a(t) - 7425.50}{(1 - B)(1 - 0.31B)}$$

其模型預測結果如下圖：

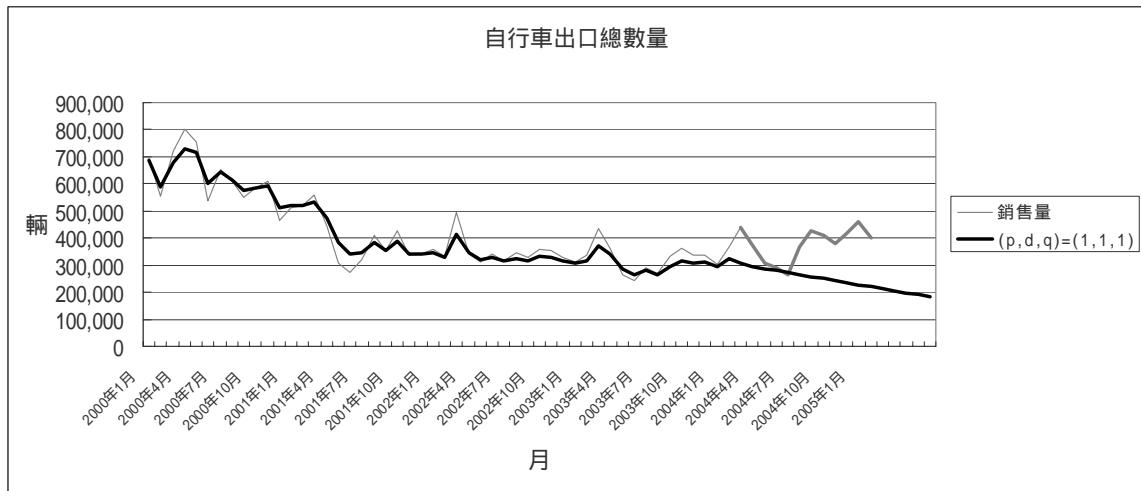


圖 5.11 需求預測模型預測結果

## 5.2.2 最佳化

實驗中利用 ILOG-CPLEX 進行求解。首先對第三章所構建之模型進行基礎實驗，因此設定單一產品，其物料、工廠、倉庫、配送中心、零售商皆為一單位，其目標式為線性規劃，在線性規劃中分別以規劃週期為 4 及 10 個時段進行求解。在損失訂單資料方面因無法獲得實際資料，因此假設未結清訂單與損失訂單之間關係，以模擬獲得未結清訂單之數值資料，其關係定義如下：

定義損失訂單佔未結清訂單的 10%， $LO(1) = 0.1 * BO(0)$

因此原式  $BO(1) = BO(0) + R(1) - d(1) - LO(1)$

可改寫為  $BO(1) = 0.9 * BO(0) + R(1) - d(1)$

以下在線性規劃中進行求解，分兩種規劃週期檢視其結果。

### 1. 線性規劃(4 個時段)

目標式考量銷售金額、存貨成本、運量成本及未結清定單成本，規劃週期(t+1)之目標式如下(自行車出口單價資料為美元，因此在成本設計上亦以美元為單位)：

$$J = 200 * \begin{bmatrix} d(0) \\ \vdots \\ d(t) \end{bmatrix} - 10 * \begin{bmatrix} y_w(0) \\ \vdots \\ y_w(t) \end{bmatrix} - 10 * \begin{bmatrix} y_D(0) \\ \vdots \\ y_D(t) \end{bmatrix} - 10 * \begin{bmatrix} y_R(0) \\ \vdots \\ y_R(t) \end{bmatrix} - 15 * \begin{bmatrix} x_{PW}(0) \\ \vdots \\ x_{PW}(t) \end{bmatrix} - 15 * \begin{bmatrix} x_{WD}(0) \\ \vdots \\ x_{WD}(t) \end{bmatrix} - 15 * \begin{bmatrix} x_{DR}(0) \\ \vdots \\ x_{DR}(t) \end{bmatrix} - 20 * \begin{bmatrix} BO(0) \\ \vdots \\ BO(t) \end{bmatrix}$$

在實驗所需虛擬數據中，存貨及運量上限假設均大於預測訂單需求(因預測出之需求量最大值為 220,455，因此各階層之存貨及運量上限均假設為 250,000)，在敏感度分析中再予以調整。因模式中考量貨運延遲之問題，存貨與貨運量皆與之前時段之貨運量有關，因此會虛擬各階層的初期運量及存貨；各階層之初始存貨假設為 0，初始運量則假設為 200,000。

以 C 語言建立  $t=0$ ,  $T_h=3$ ,  $L=1$  之模型，圖 5.12 為程式示意圖，模式包含 31 個變數，32 條限制式，經由呼叫 CPLEX 進行最佳化求解，圖 5.13 為 CPLEX 求解結果示意圖，輸出各變數經由最佳化所獲得之解，表 5.9 為各變數在各時段的最佳化結果，圖 5.14 為決策結果路網圖，利用路網圖說明時段中各階層的運作情形，在圖 5.14 中發現存貨數皆為 0，工廠僅在  $t=0$  時生產出貨 228,685 單位。

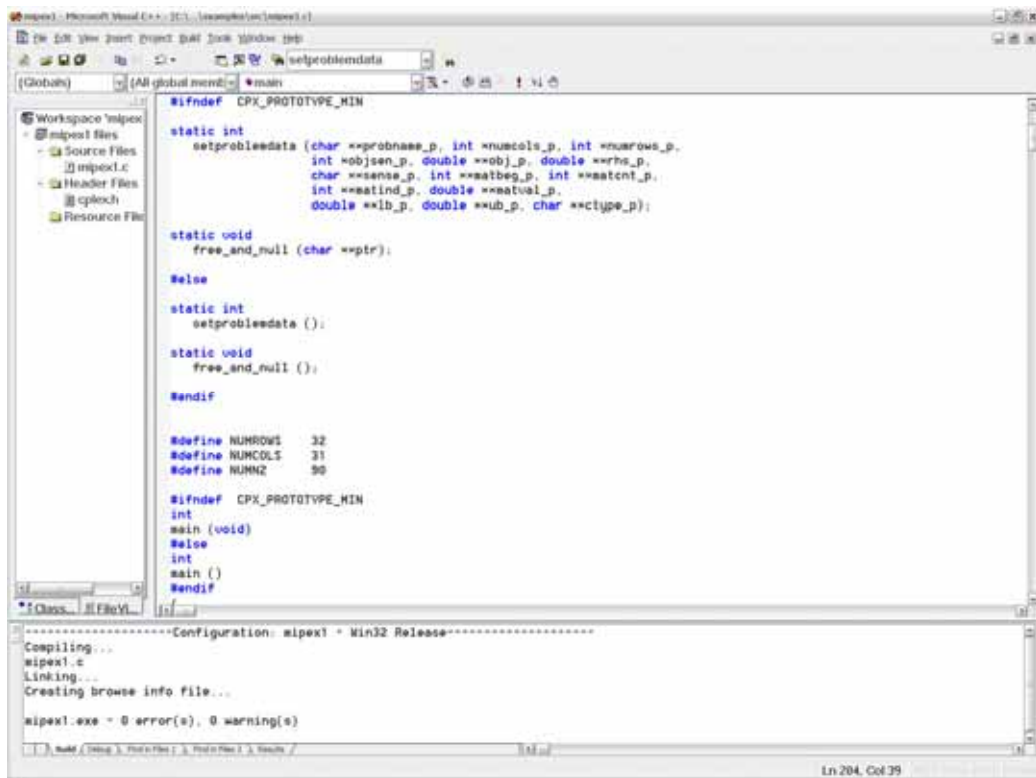


圖 5.12 程式示意圖

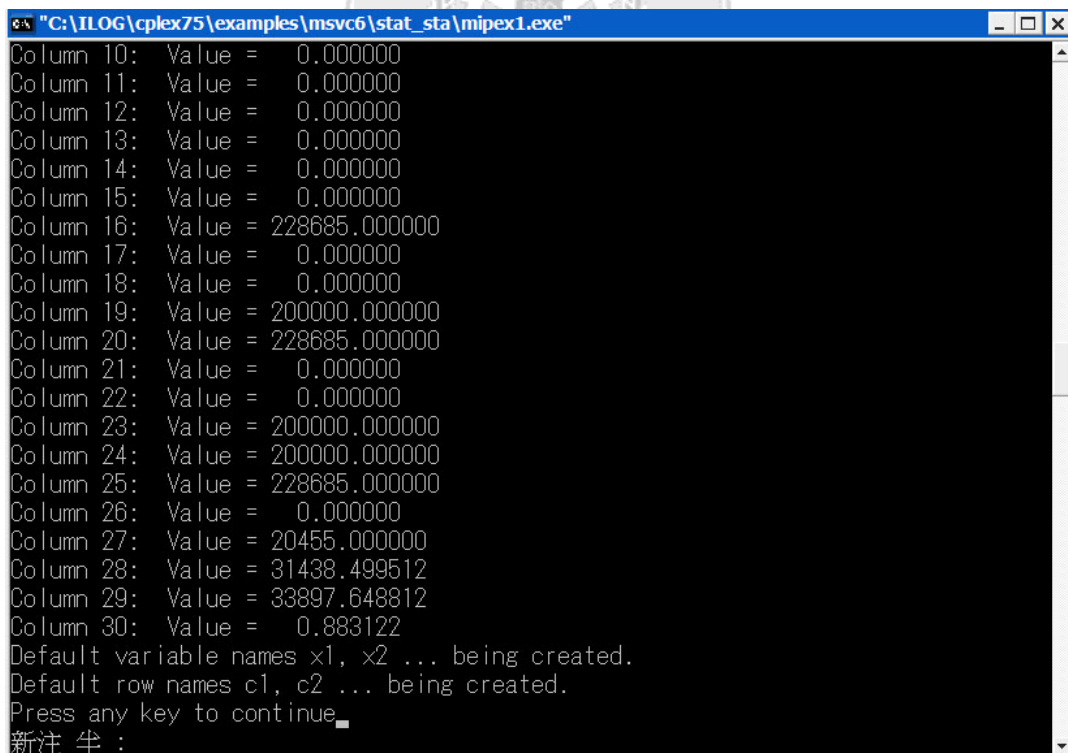


圖 5.13 CPLEX 求解結果示意圖

表 5.9 線性規劃下最佳化 4 個時段之結果

t \	0	1	2	3
訂單	220,455	213,029	205,603	198,178
d	200,000	200,000	200,000	228,685
$y_W$	0	0	0	0
$y_D$	0	0	0	0
$y_R$	0	0	0	0
$x_{PW}$	228,685	0	0	-
$x_{WD}$	200,000	228,685	0	0
$x_{DR}$	200,000	200,000	228,685	0
BO	20455	31438	33897	0

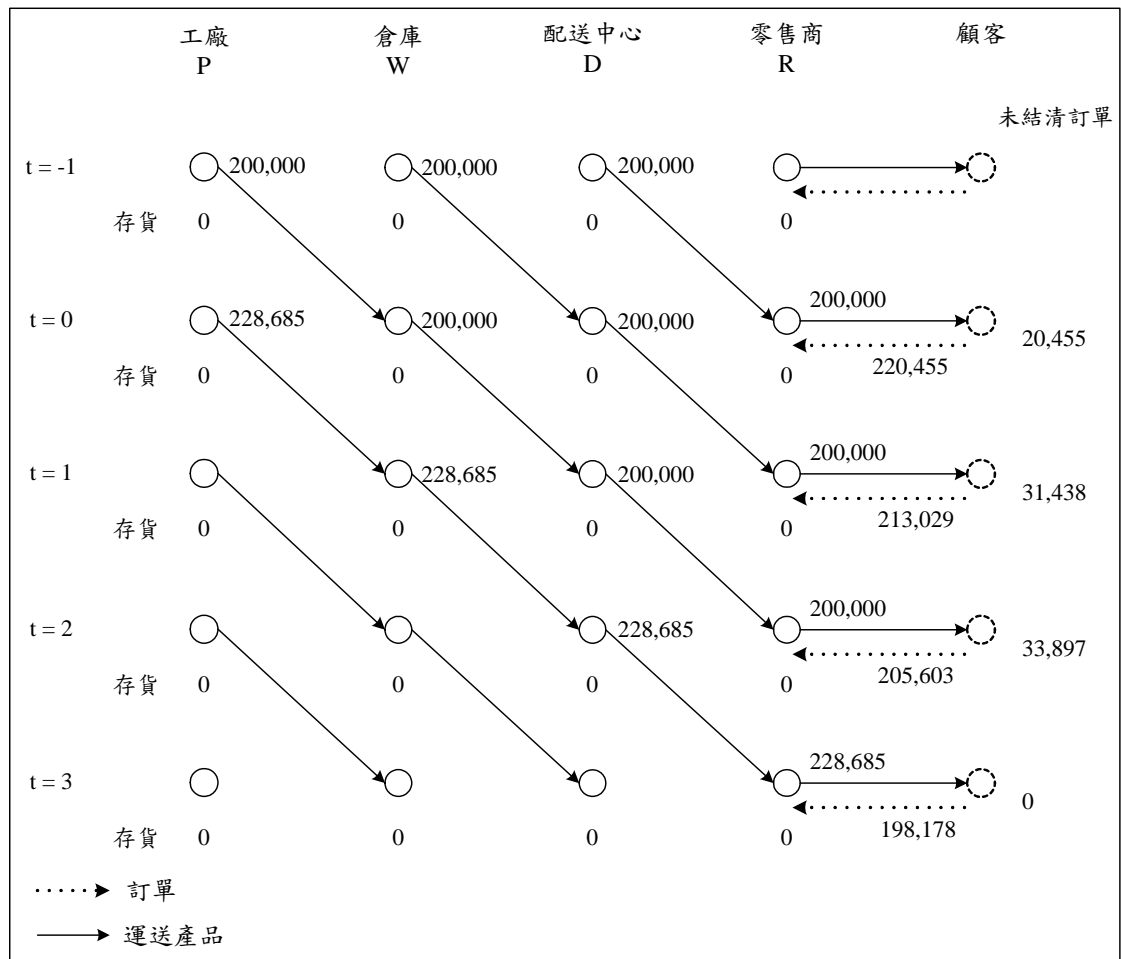


圖 5.14 決策結果路網圖(4 個時段)



由決策結果發現以下情形：

- 因各階層初期均有供貨至下游階層，且工廠至倉庫之運量上限充足，因此計算零售商所短缺的產品數量後，由工廠於第 0 時段將所需產品數量運至下游。

以下進行敏感度分析，藉由不同情境進行最佳化產生決策檢視模式。

### 敏感度分析

以前面實驗為基礎，設計兩種情境進行最佳化；情境一是檢視產品銷售單價依時段不同時最佳化之情形；情境二則是檢視在工廠-倉庫運量限制下的最佳化結果。

#### 情境一、銷售單價

實驗中發現產品單價原為 200 元(美金)，當  $t=3$  時產品單價若漲為 229 元(美金)則小幅度影響前一期  $t=2$  的銷售量，為明顯顯示其影響將產品在  $t=3$  時之單價漲為 260 元(美金)，檢視其最佳化結果。結果如圖 5.15，在  $t=2$  時銷售量僅剩下 2 單位；由決策結果發現以下情形：

- 在預期價格上漲的考量下，前一期發生銷售量遽減之情形。
- 因產品延後出貨導致未結清訂單( $t=2$ )增加，相對增加損失訂單，因此總銷售量下降。
- 因配送中心-零售商運量上限為 250,000 單位，因此零售商需預先存貨以能在產品單價高的時段大量銷售；配送中心受限於上游運量及下游訂單亦產生存貨。

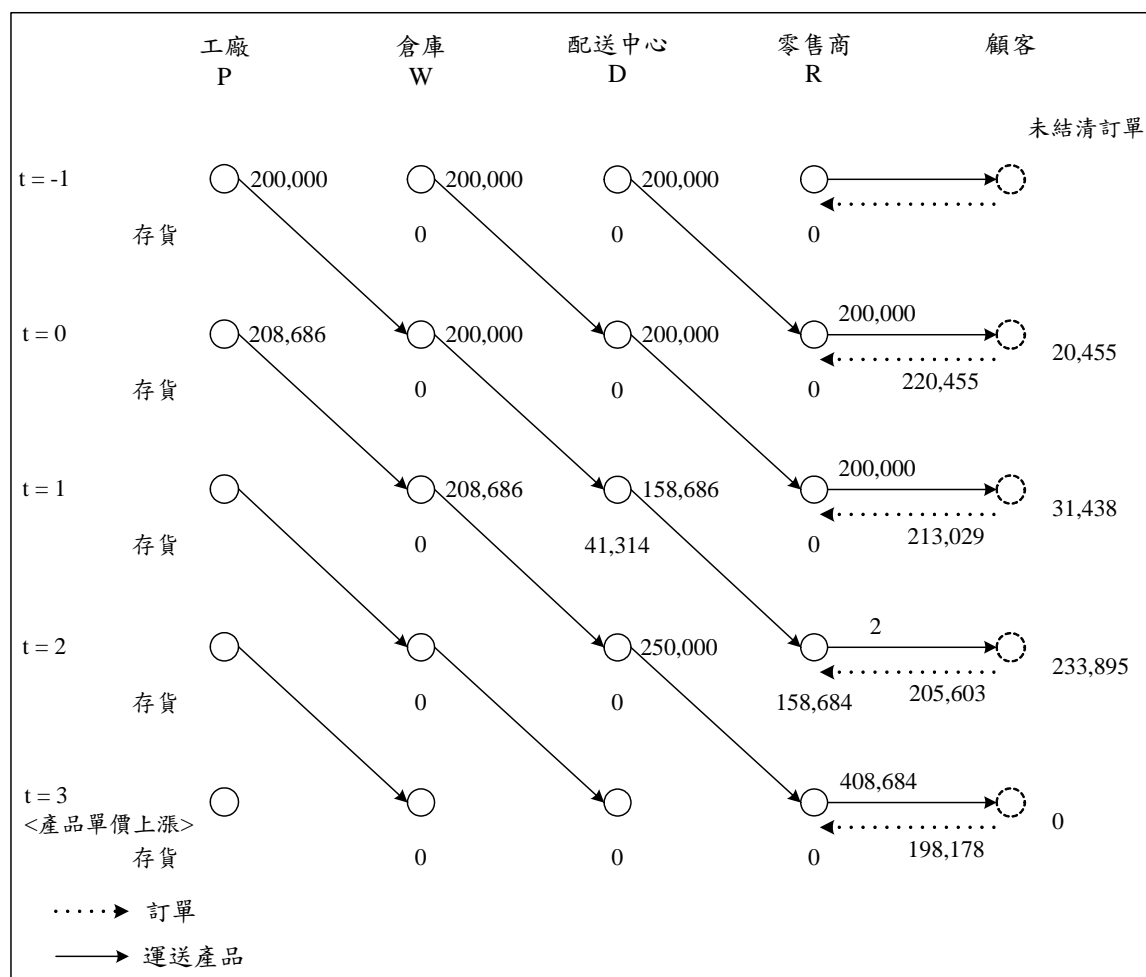


圖 5.15 決策結果路網圖(4 個時段.情境一)

## 情境二、工廠-倉庫運量上限

為檢視工廠-倉庫運量限制對模式的影響，將工廠-倉庫運量上限減低成 200,000 單位使運量不足檢視最佳化結果。結果如圖 5.16，工場僅在 t=0 時生產出貨 200,000；由決策結果發現以下情形：

- 因規劃週期過短，工廠僅在 t=0 時運送至下游產品能到達零售商，因此如限制工廠-倉庫運量過低時，會產生供貨不足、未結清定單累積之情形。

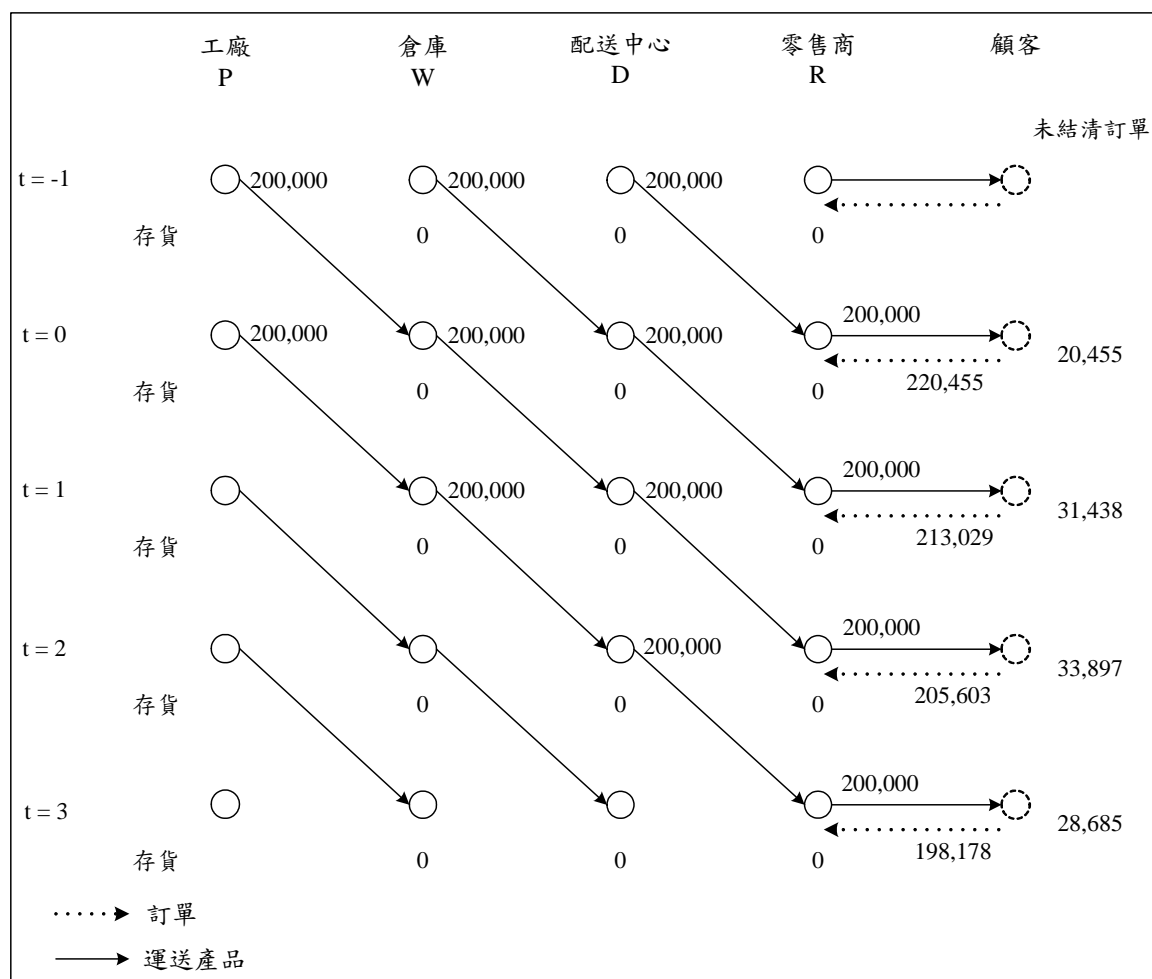


圖 5.16 決策結果路網圖(4 個時段.情境二)

### 目標式結果討論

將最佳化後目標式結果歸納如表 5.10，有下列幾點發現：

- 在無情境設計下其利潤及銷售均最高。
- 情境一則因改變某一時段之產品銷售單價，因此發生存貨成本，而由於延後出貨導致損失訂單增加，實際銷售量降低相對銷售額降低，因此總利潤比無情境之利潤為低。
- 情境二受限於運量限制及最佳化長度不足，因此其利潤為低。

表 5.10 多階層單一產品最佳化目標式結果(4 個時段)

			情境一 (價格上漲)	情境二 (運量限制)
利潤		144,730,334	135,630,565	139,710,459
收益	銷售量	165,737,000	161,737,200	160,000,000
存貨成本	工廠	0	0	0
	配送中心	0	413,140	0
	零售商	0	1,586,840	0
運量成本	工廠-倉庫	3,430,275	3,130,290	3,000,000
	倉庫-配送中心	6,430,275	6,130,290	6,000,000
	配送中心-零售商	9,430,275	9,130,290	9,000,000
懲罰成本	未結清訂單	1,715,840	5,715,784	2,289,540

## 2. 線性規劃(10 個時段)

在最佳化 4 個時段時發現，因供應鏈階層數及前置時間下規劃週期 4 明顯過短，因此將時段延伸至 10 個時段進行最佳化。

實驗所需虛擬數據與最佳化 4 個時段的數值實驗假設相同。存貨及運量上限假設均大於預測訂單需求(因預測出之需求量最大值為 220,455，因此各階層之存貨及運量上限均假設為 250,000)，在敏感度分析中再予以調整。因模式中考量貨運延遲之問題，存貨與貨運量皆與之前時段之貨運量有關，因此會虛擬各階層的初期運量及存貨；各階層之初始存貨假設為 0，初始運量則假設為 200,000。

以 C 語言建立  $t=0$ ,  $T_h=9$ ,  $L=1$  之模型，經最佳化求得結果如圖 5.17，存貨量皆為 0，工廠從  $t=0$  開始之產量與 3 個時段之後零售商之訂單數相同，未結清訂單從  $t=3$  開始為 0；由決策結果發現以下情形：

- 由於無初期存貨，初期運量亦不足以供應需求，因此產生未結清訂單。

- 工廠依零售商預測需求量加上未結清訂單數量生產所需產品，而工廠生產之產品運至零售商需 3 個時段，因此  $t=0$  時運送出，在  $t=3$  時運送至零售商，將未結清訂單此時才處理完畢。
- 在  $t=3$  時段之後，各階層間之運量皆是以需求量往回推算決定，發現初期存貨及運量之假設影響極大。

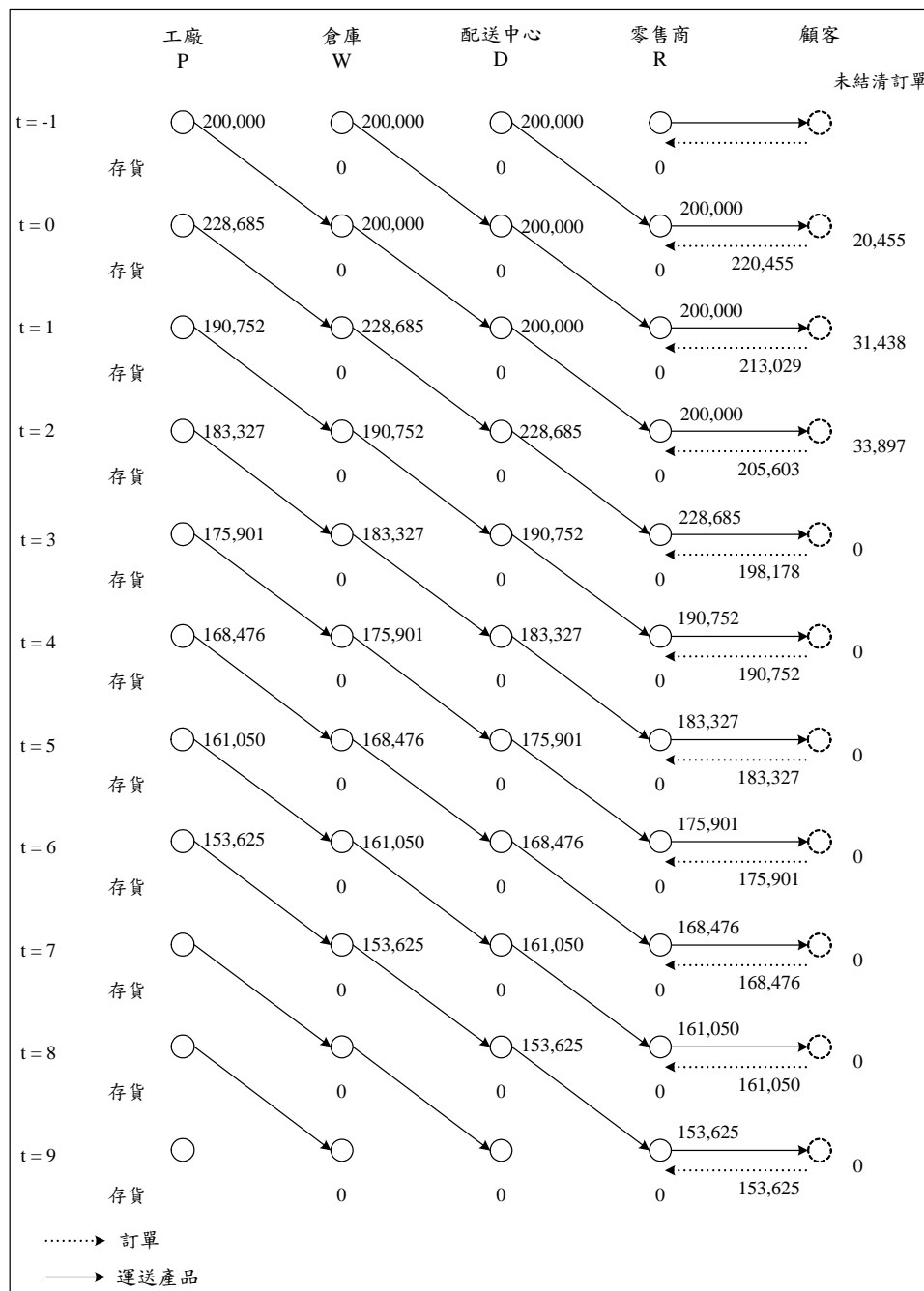


圖 5.17 決策結果路網圖(10 個時段)

以下將進行敏感度分析，藉由不同情境進行最佳化產生決策檢視模式。

### 敏感度分析

在最佳化 4 個時段實驗中發現，在進行工廠-倉庫運量限制敏感度分析時最佳化長度不夠長，因此在最佳化 10 個時段實驗中針對這一部分再進行一次敏感度分析；依循情境一的最佳化結果設計情境二，改變存貨成本以檢視存貨成本對最佳化結果之影響。

#### 情境一、工廠-倉庫運量

為檢視工廠-倉庫運量限制對模式的影響，將  $t=1$  時工廠-倉庫的運量上限減低成 150,000 單位使運量不足檢視最佳化結果。結果如圖 5.18，工廠在  $t=0$  及  $t=2$  皆增加產能運出，零售商在  $t=3$  時存貨 21,315 單位，未結清訂單在  $t=3$  時為 0，但在  $t=4$  時產生 19,437 單位；由決策結果發現以下情形：

- 最佳化結果中在前一期  $t=0$  會預先增加運量，但礙於運量上限仍無法預先補足運量，因此  $t=2$  的運量也會增加以補足因限制加大而不足的運量。
- 因預先增加運量，零售商於  $t=3$  時產生存貨量。

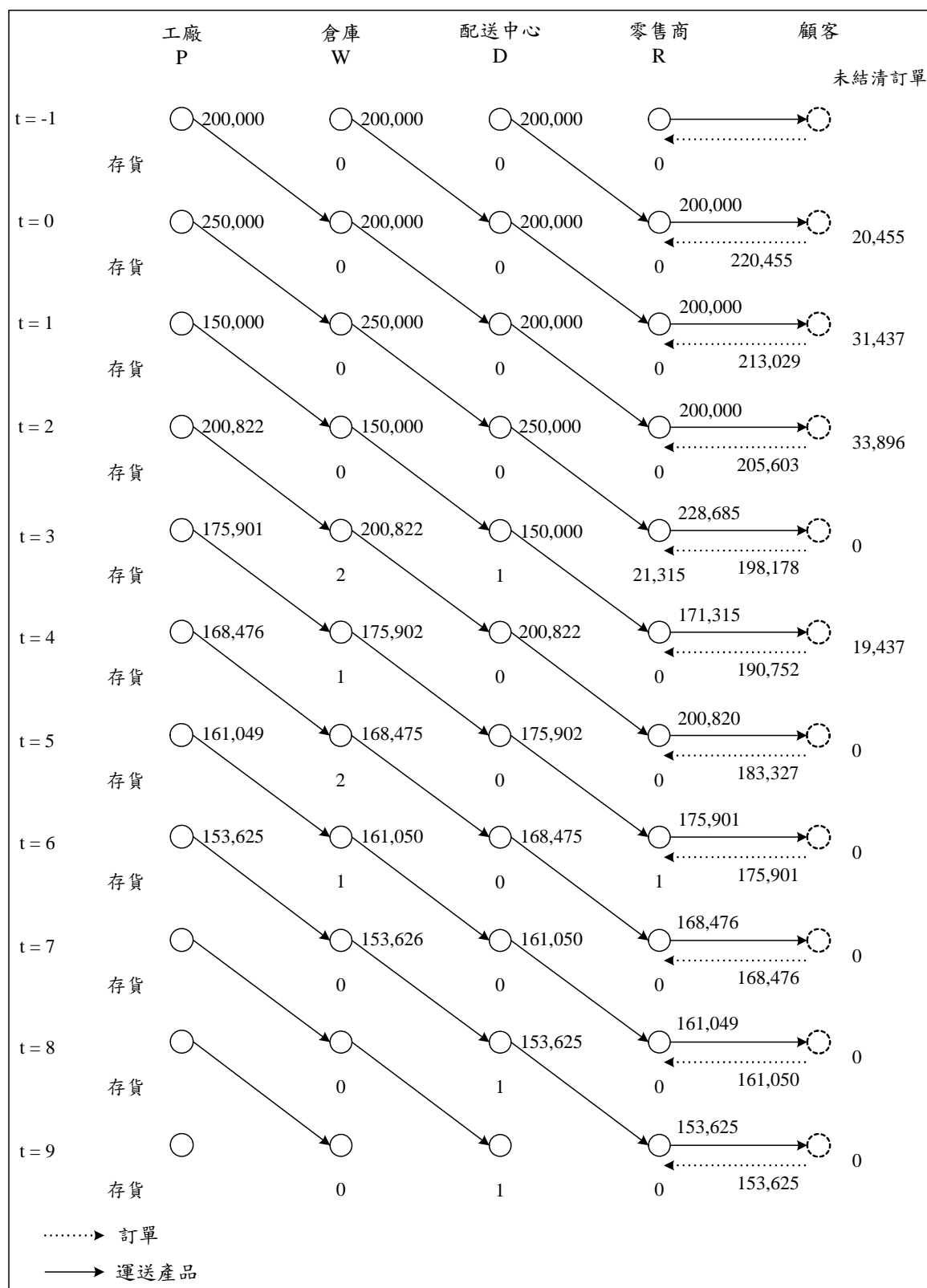


圖 5.18 決策結果路網圖(10 個時段.情境一)

情境二、存貨成本

因情境一產生之最佳化結果中零售商產生存貨，此時各階層之單位存貨成本皆為 10 元(美金)，因此將零售商之單位存貨成本調整成 11 元(美金)，檢視最佳化結果。結果如圖 5.19，倉庫在  $t=1$  時存貨 21,315 單位；由決策結果發現以下情形：

- 因零售商存貨成本上升，因此決策改為在倉庫發生存貨情形。

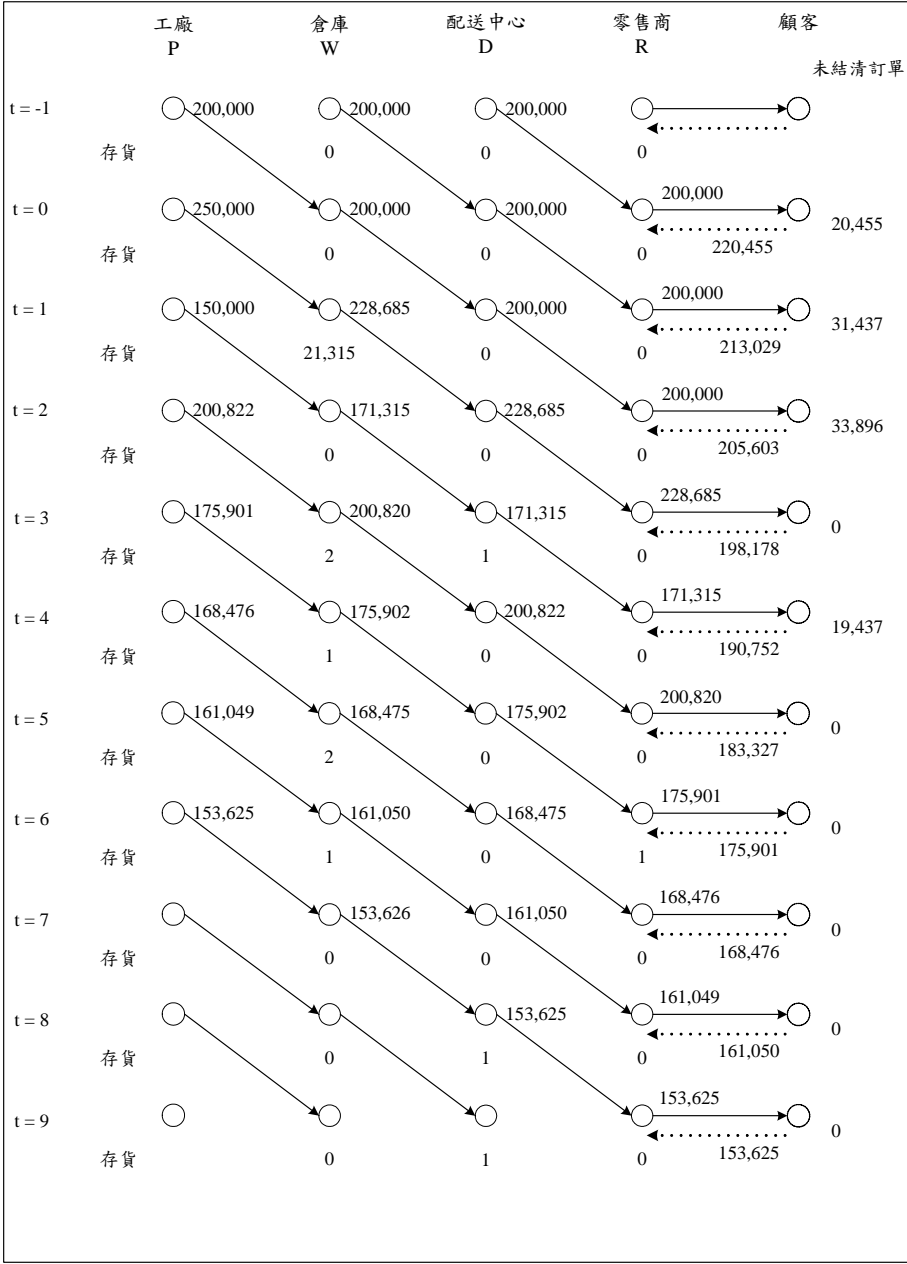


圖 5.19 決策結果路網圖(10 個時段.情境二)



## 目標式結果討論

將最佳化後目標式結果歸納如表 5.11，有幾點發現：

- 在運量上限、存貨上限、物料量皆充足情形下其利潤及銷售均最高。
- 情境一則因發生運量限制的事件因此工廠-倉庫運量成本較小，但銷售額及未結清訂單增加，導致利潤下降。
- 情境二在零售商存貨成本調高的情形下變更負擔存貨的階層，加總存貨成本可以發現存貨總成本並無增加，其利潤與情境一相同。

表 5.11 多階層單一產品最佳化目標式結果(10 個時段)

			情境一 (運量限制)	情境二 (零售商存貨 成本增加)
利潤		304,865,695	303,962,340	303,962,340
收益	銷售額	372,363,200	371,974,400	371,974,400
存貨成本	倉庫	0	60	213,210
	配送中心	0	20	20
	零售商	0	213,160	11
運量成本	工廠-倉庫	18,927,240	18,898,095	18,898,095
	倉庫-配送中心	21,927,240	21,898,095	21,898,095
	配送中心-零售商	24,927,240	24,898,080	24,898,080
懲罰成本	未結清訂單	1,715,784	2,104,549	2,104,549

### 5.2.3 結果討論

在多階層單一產品的數值實驗中比較不同規劃週期長度，以及不同型態之目標式所獲得之決策結果，獲得初步分析如下：

1. 最佳化時段數之決定應考量前置時間及供應鏈階層數。

2. 在最佳化時段數拉長後發現決策產生規律性，並且無存貨發生，直接由進貨直接提供出貨。
3. 在產品單價上漲的情形下決定延後售出未必能為供應鏈帶來較大的利潤。
4. 運量限制導致供貨不足如為事件性發生，非連續發生則可由前一時段或後一時段增加運量解決運量不足的情形；但如持續連續發生則會導致未結清訂單持續累積無法處理完畢。
5. 在存貨成本不同時模式會予以判斷存貨成本該由哪一階層負擔。

### 5.3 多階層多產品實驗

在多階層單一產品的數據實驗結果中可以檢視模式在不同情境下之決策反應，本節將延伸多階層單一產品成多階層單多產品，檢視 5.1 節所設計之案例在不同情境下之決策反應。以下先建立需求預測模式，再利用預測獲得之需求量進行最佳化獲得決策。

#### 5.3.1 需求預測模型

經由需求預測模式可以預測未來需求以提供最佳化決策之用，配合案例設計，以下將在 ARIMA 模式基礎下以不同歷史資料筆數針對 5 筆需求資料進行模式建立。實驗數據以自行車出口國中選取前三大出口國之數據資料，電動自行車出口國中則選取二大出口國之數據資料進行需求預測模式之建立。

利用不同筆數之歷史資料預測未來 12 筆需求資料，以均方根差 (root mean squared error, RMSE) 與實際資料作一比對，選取適合需求特性之滾動方式，結果如表 5.12 至表 5.16，發現在不同資料型態下不同歷史資料筆數所獲得之預測結果確實有所不同，將五種資料的最佳歷史資料筆數歸納成表 5.17，變動週期較長或是無週期之資料所需之歷史資料筆數較多，反之週期較短之資料則以較少之歷史資料筆數進行預測較佳；此外將選取之模式預測結果與實際資料作一趨勢圖比較，如圖 5.20 至圖 5.24，預測模式確實能預測出資料之趨勢：

表 5.12 自行車出口至北美自由貿易區(NAFTA)

歷史資料筆數	實際資料	*12	24	36	48
模式階數(p,d,q)		(2,1,1)	(1,2,1)	(1,1,1)	(1,1,1)
1	89,722	93,004	66,460	68,391	57,683
2	80,075	105,899	65,354	67,732	50,249
3	72,955	95,554	64,325	66,501	44,280
4	74,634	78,228	63,378	65,073	39,253
5	69,300	74,206	62,514	63,577	34,831
6	107,080	85,491	61,734	62,057	30,798
7	109,948	97,467	61,036	60,529	27,016
8	84,759	97,811	60,421	58,999	23,395
9	81,258	89,063	59,890	57,468	19,877
10	86,639	82,863	59,441	55,936	16,426
11	80,546	85,747	59,075	54,404	13,018
12	78,196	93,279	58,792	52,872	9,638
RMSE		<b>13985.864</b>	25948.561	26986.711	57444.234

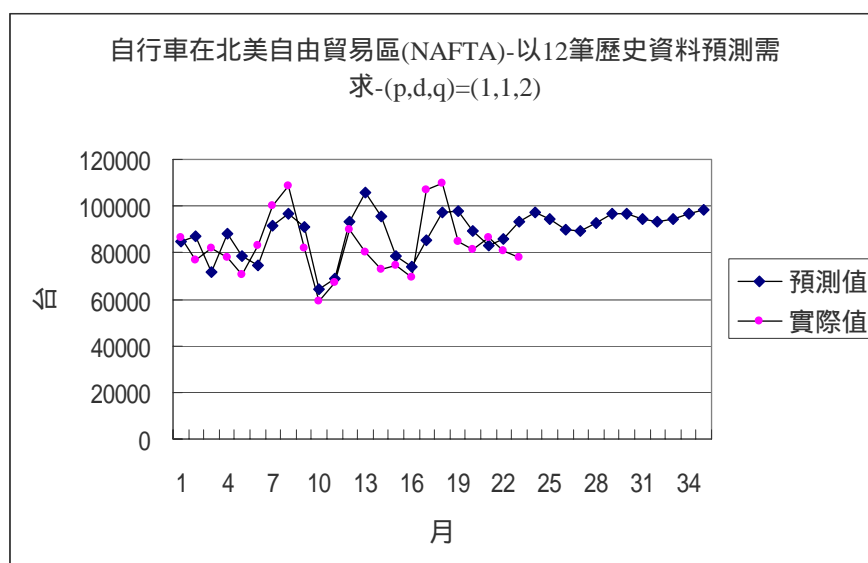


圖5.20 自行車出口至北美自由貿易區預測資料與實際資料趨勢圖

表 5.13 自行車出口至歐盟(EU)

歷史資料筆數	實際資料	*12	24	36	48
模式階數(p,d,q)		(1,1,0)	(1,1,1)	(0,1,1)	(1,1,1)
1	240,753	208,317	196,472	178,115	185,667
2	198,155	208,200	188,370	180,051	169,712
3	180,963	208,083	184,675	181,987	160,308
4	157,240	207,966	183,304	183,923	154,695
5	131,610	207,850	183,157	185,859	151,276
6	203,662	207,733	183,656	187,794	149,129
7	252,592	207,616	184,495	189,730	147,717
8	254,579	207,499	185,513	191,666	146,731
9	211,913	207,382	186,626	193,602	145,991
10	258,295	207,266	187,788	195,538	145,394
11	303,027	207,149	188,977	197,474	144,879
12	277,387	207,032	190,179	199,409	144,412
RMSE		<b>51069.484</b>	58763.456	55949.635	86627.974

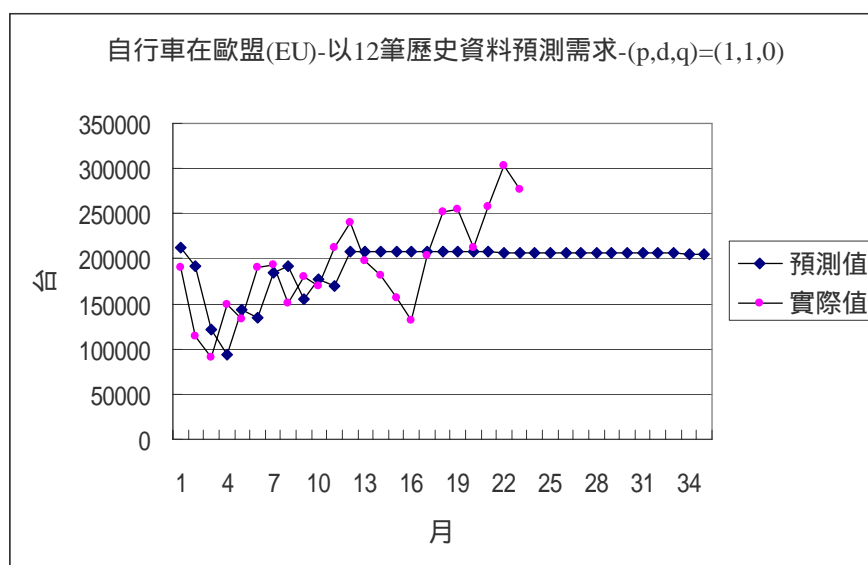


圖5.21 自行車出口至歐盟預測資料與實際資料趨勢圖

表 5.14 自行車出口至歐協(EFTA)

歷史資料筆數	實際資料	12	24	36	*48
模式階數(p,d,q)		(0,2,2)	(0,2,1)	(0,2,1)	(3,1,1,)
1	19,228	37,272	35,097	34,593	24,327
2	18,963	50,596	43,102	41,795	27,146
3	4,804	65,968	51,906	49,497	23,289
4	4,668	83,390	61,507	57,698	17,101
5	2,916	102,861	71,908	66,398	11,283
6	4,348	124,381	83,106	75,597	6,783
7	4,072	147,951	95,103	85,296	4,729
8	10,124	173,569	107,898	95,494	4,863
9	11,867	201,237	121,492	106,191	6,782
10	19,695	230,954	135,883	117,387	9,474
11	27,343	262,720	151,074	129,083	12,131
12	18,651	296,536	167,062	141,278	14,016
RMSE		157106.873	90344.337	77667.989	<b>9471.533</b>

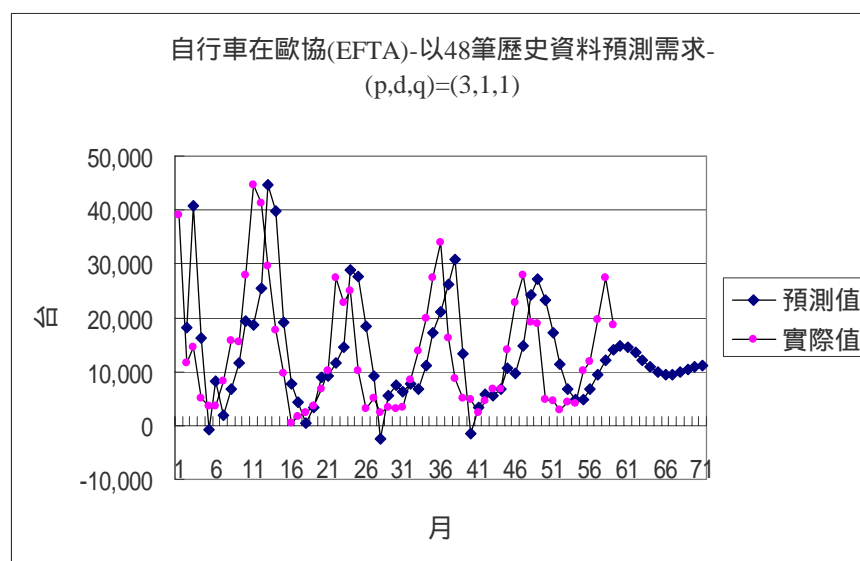


圖5.22 自行車出口至歐協預測資料與實際資料趨勢圖

表5.15 電動自行車出口至北美自由貿易區(NAFTA)

歷史資料筆數	實際資料	12	24	36	*48
模式階數(p,d,q)		(1,0,1)	(1,1,2)	(1,1,1)	(1,1,1)
1	198	521	-649	557	540
2	354	838	-45	802	771
3	167	1,008	-298	964	921
4	493	1,100	-432	1,069	1,016
5	155	1,150	-583	1,137	1,075
6	233	1,177	-732	1,180	1,110
7	335	1,191	-881	1,207	1,129
8	587	1,199	-1,030	1,222	1,136
9	433	1,203	-1,180	1,231	1,137
10	452	1,206	-1,329	1,234	1,133
11	466	1,207	-1,478	1,234	1,126
12	155	1,207	-1,627	1,232	1,117
RMSE		775.705	1298.926	780.950	<b>707.272</b>

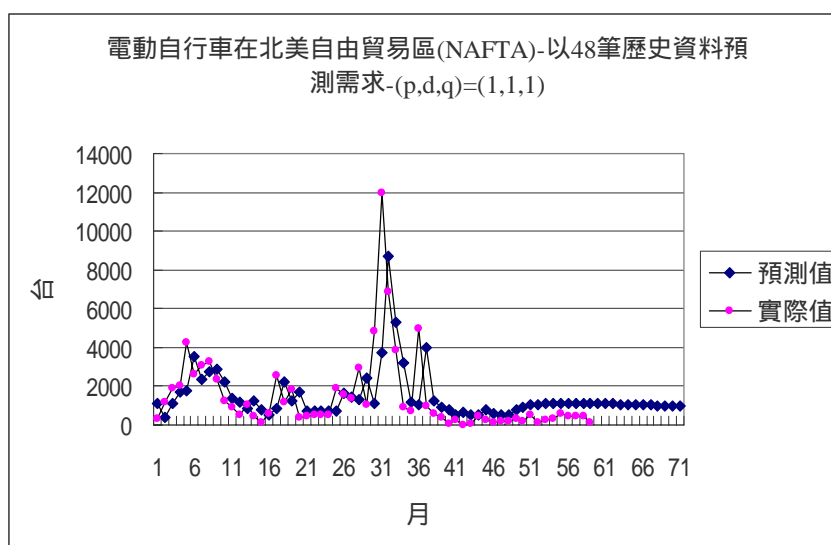


圖5.23 電動自行車出口至北美自由貿易區預測資料與實際資料趨勢圖

表5.16 電動自行車出口至歐盟(EU)

歷史資料筆數	實際資料	12	*24	36	48
模式階數(p,d,q)		(0,1,2)	(3,0,2)	(1,0,0)	(1,1,1)
1	202	636	350	529	442
2	151	654	175	479	400
3	39	704	112	445	386
4	23	753	90	421	378
5	10	803	100	405	372
6	203	852	120	393	366
7	125	902	144	385	361
8	70	952	166	380	355
9	78	1,001	184	376	350
10	61	1,051	197	374	344
11	174	1,100	207	372	339
12	119	1,150	213	371	333
RMSE		795.819	<b>89.678</b>	313.889	271.975

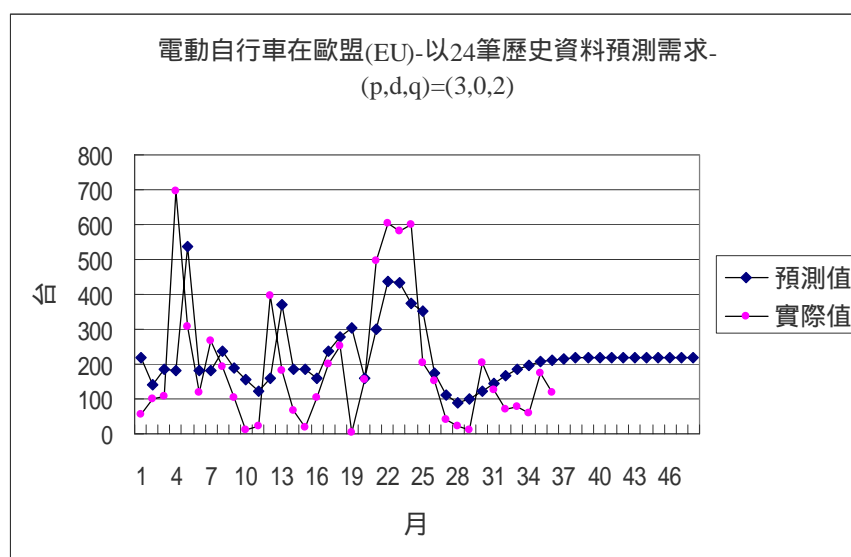


圖 5.24 電動自行車出口至歐盟預測資料與實際資料趨勢圖

表 5.17 出口資料預測之最佳歷史資料筆數

	歷史資料筆數
自行車出口至北美自由貿易區	12
自行車出口至歐盟	12
自行車出口至歐協	48
電動自行車出口至北美自由貿易區	48
電動自行車出口至歐盟	24

完成各預測模式模型如下：

1. 自行車出口至北美自由貿易區

$$Z(t) = \frac{(1 + 0.021B - 0.931B^2)a(t) - 483.546}{(1 - B)(1 - 0.259B)}$$

2. 自行車出口至歐盟

$$Z(t) = \frac{(1 - 0.106B)a(t) - 116.812}{(1 - B)}$$

3. 自行車出口至歐協

$$Z(t) = \frac{(1 - 0.961B)a(t) - 73.660}{(1 - B)(1 - 0.823B - 0.285B^2 + 0.539B^3)}$$

4. 電動自行車出口至北美自由貿易區

$$Z(t) = \frac{(1 - 0.984B)a(t) - 13.320}{(1 - B)(1 - 0.666B)}$$

5. 電動自行車出口至歐盟

$$Z(t) = \frac{(1 - 0.423B - 0.574B^2)a(t) + 217.533}{(1 - B)(1 - 0.911B - 0.094B^2 + 0.207B^3)}$$



由實驗結果發現，不同之資料型態需要不同週期的預測方式，而 ARIMA 模式確實能預測出需求資料未來之趨勢。

### 5.3.2 最佳化

本研究以自行車產業為案例設計對象進行範例分析，所設計之架構主要是以公司為主體，在公司能獲得上下游資訊下考量整體效益進行決策分析，但因受限於無法取得實際營運決策資料，因此僅以整體產業架構設計案例。自行車產業是以每季檢視其營運狀況，因此將半年(兩季)分為六個時段進行最佳化求解。依據 5.1 節案例設計構建模式，在原物料運送至零件商時會先行存貨再製成零件，因此零件商包含物料及零件存貨；在零件製成後運送至裝配廠亦是先行存貨再裝配成產品，因此裝配廠包含零件及產品存貨；在產品生產完成後運至經銷商，由經銷商依顧客對產品之不同需求運送出產品。

與多階層單一產品的不同除了產品種類數不同外，時段的考量亦不同，在多階層單一產品中是考量每一變數由  $t=0$  到  $t=T_h$ ，因此須虛擬一些初期數據如初期運量，且有些變數會因此成為無效變數，如後期工廠-倉庫之運量因在最佳化週期內無法運至零售商其運量會為 0；而多階層多產品的變數考量則是依前置時間或生產時間往前定義變數，因此以第一期訂單為  $t=0$  往前回推上游變數所在之時段，各變數時段說明如圖 5.25。

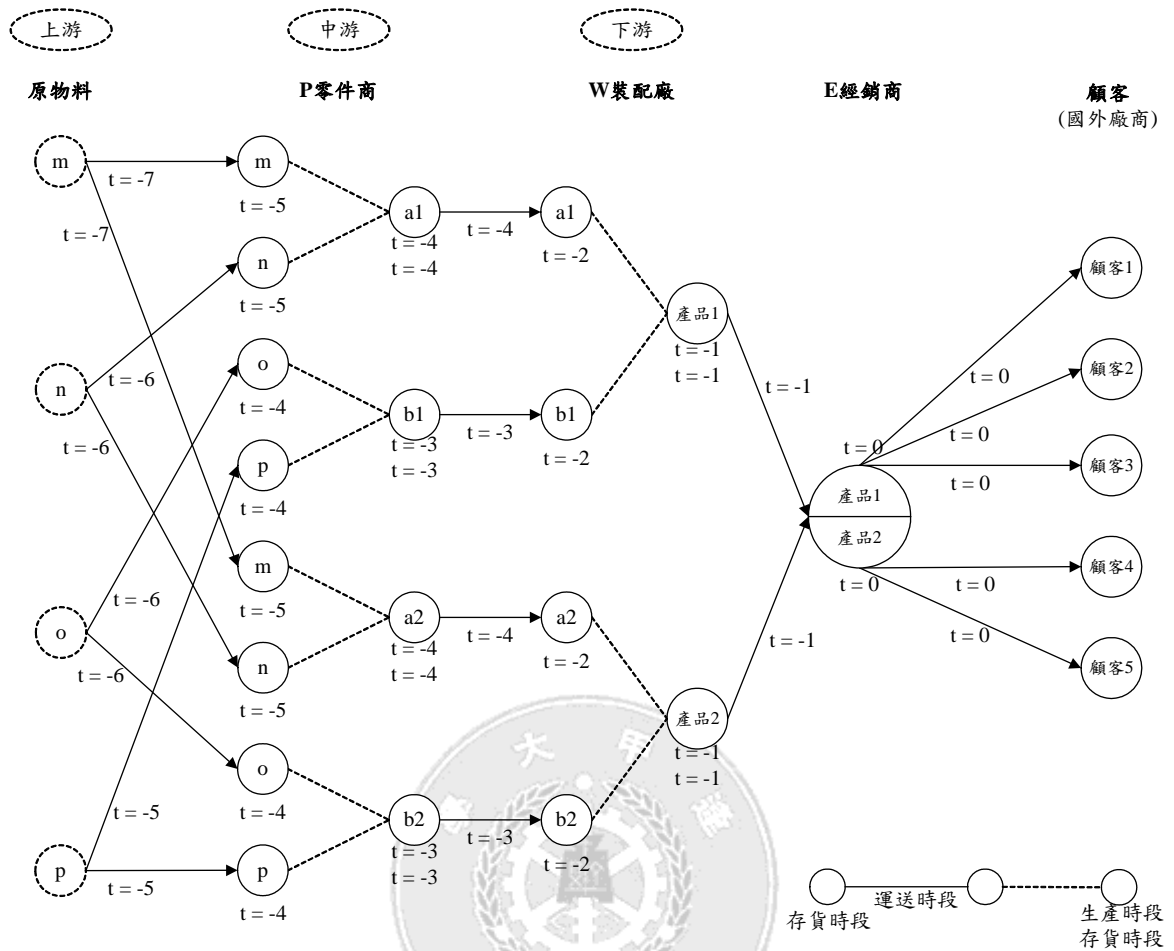


圖 5.25 變數時段說明

模式變數說明及目標式係數如表 5.18 所示，6 個時段共有 300 個變數。經最佳化求得結果如表 5.19 所示，因存貨結果皆為零，而在敏感度分析結果中僅少數階層在某些時段不為零，因此省略存貨皆為零的部份，完整結果請參見附錄二。

表 5.18 變數說明、目標式係數

存貨													
P 零件商													
變數	物料 (a1(m))	物料 (a1(m))	a1	物料 (b1 (0))	物料 (b1 (p))	b1	物料 (a2 (m))	物料 (a2 (n))	a2	物料 (b2 (o))	物料 (b2 (p))	b2	
時段	x1 - x6 (-5) - (0)	x7 - x12 (-5) - (0)	x13 - x18 (-4) - (1)	x19 - x24 (-4) - (1)	x25 - x30 (-4) - (1)	x31 - x36 (-3) - (2)	x37 - x42 (-5) - (0)	x43 - x48 (-5) - (0)	x49 - x54 (-4) - (1)	x55 - x60 (-4) - (1)	x61 - x66 (-4) - (1)	x67 - x72 (-3) - (2)	
目標式係數	-10	-5	-12	-8	-7	-10	-10	-5	-12	-8	-7	-10	

存貨									
W 裝配廠									
零件 a1 (產品 1)		零件 b1 (產品 1)	產品 1	零件 a2 (產品 2)	零件 b2 (產品 2)	產品 2	產品 1	產品 2	
x73 - x78 (-2) - (3)	x79 - x84 (-2) - (3)	x85 - x90 (-1) - (4)	x91 - x96 (-2) - (3)	x97 - x102 (-2) - (3)	x103 - x108 (-1) - (4)	x109 - x114 (0) - (5)	x115 - x120 (0) - (5)	-20	
-12	-8	-10	-12	-8	-10	-15			

運量									
原物料 - 零件商									
m - a1	m - a2	n - a1	n - a2	o - b1	o - b2	p - b1	p - b2		
x121 - x126 (-7) - (-2)	x127 - x132 (-7) - (-2)	x133 - x138 (-6) - (-1)	x139 - x144 (-6) - (-1)	x145 - x150 (-6) - (-1)	x151 - x156 (-6) - (-1)	x157 - x162 (-5) - (0)	x163 - x168 (-5) - (0)	-5	
-10	-7	-10	-7	-7	-5	-7			

產量																	
零件商產量						裝配廠產量											
a1		b1		a2		b2		產品 1	產品 2								
x169	-	x174	x175	-	x180	x181	-	x186	x187	-	x192	x193	-	x198	x199	-	x204
(-4)	-	(1)	(-3)	-	(2)	(-4)	-	(1)	(-3)	-	(2)	(-1)	-	(4)	(-1)	-	(4)
-20			-10			-30			-10					-10		-20	

運量				銷售量							
零件商 - 裝配廠				裝配廠 - 經銷商		經銷商 - 顧客					
a1 - 產品 1		b1 - 產品 2	a2 - 產品 1	產品 1 - 經銷商	產品 2 - 經銷商	產品 1 - C1	產品 1 - C2	產品 1 - C3	產品 2 - C1	產品 2 - C2	
x205 - x210	(-1)	(-3)	(-2)	(-4)	(-1)	(-4)	(-1)	(-4)	(-1)	(-4)	(-1)
-10		-5	-15	-5	-15	-20	290	161	239	559	500

\*銷售單價取決於自行車及電動自行車前兩季出口至各國之平均出口單價

未結清訂單														
經銷商														
產品 1 - C1		產品 1 - C2	產品 1 - C3	產品 2 - C1	產品 2 - C2									
x271	-	x276	x277	-	x282	x283	-	x288	x289	-	x294	x295	-	x299
(0)	-	(5)	(0)	-	(5)	(0)	-	(5)	(0)	-	(5)	(0)	-	(5)
-50		-50		-50										
				-70									-70	

表 5.19 多階層多產品最佳化結果

			情境一	情境二
無運量限制、無初期存貨			物料不足	產能不足
變數	時段		$X_{1.2.3.4.5.6} < 280,000$	$X_{194.195} < 150,000$ $X_{200.201} < 500$
求解結果			求解結果	求解結果
x1	(-5)	0	0	0
x84	(3)	0	0	0
x85	(-1)	0	0	0
x86	(0)	0	0	350000
x87	(1)	0	0	203079
x88	(2)	0	0	0
x89	(3)	0	0	0
x90	(4)	0	0	0
x91	(-2)	0	0	0
x96	(3)	0	0	0
x97	(-2)	0	0	0
x102	(3)	0	0	0
x103	(-1)	0	0	0
x104	(0)	0	0	1625
x105	(1)	0	0	806
x106	(2)	0	0	0
x107	(3)	0	0	0
x108	(4)	0	0	0
x109	(0)	0	0	0

x110	(1)	0	0	54672
x111	(2)	0	0	0
x112	(3)	0	0	0
x113	(4)	0	0	0
x114	(5)	0	0	0
x115	(0)	0	0	0
x120	(5)	0	0	0
x121	(-7)	354725	278658	354725
x122	(-6)	354006	278669	758678
x123	(-5)	351593	278681	150000
x124	(-4)	353079	278694	150000
x125	(-3)	358830	278707	358830
x126	(-2)	364674	278720	364674
x127	(-7)	1342	1342	1342
x128	(-6)	1331	1331	2956
x129	(-5)	1319	1319	500
x130	(-4)	1306	1306	500
x131	(-3)	1293	1293	1293
x132	(-2)	1280	1280	1280
x133	(-6)	354725	278658	354725
x134	(-5)	354006	278669	758678
x135	(-4)	351593	278681	150000
x136	(-3)	353079	278694	150000
x137	(-2)	358830	278707	358830
x138	(-1)	364674	278720	364674
x139	(-6)	1342	1342	1342
x140	(-5)	1331	1331	2956
x141	(-4)	1319	1319	500

x142	(-3)	1306	1306	500
x143	(-2)	1293	1293	1293
x144	(-1)	1280	1280	1280
x145	(-6)	354725	278658	354725
x146	(-5)	354006	278669	758678
x147	(-4)	351593	278681	150000
x148	(-3)	353079	278694	150000
x149	(-2)	358830	278707	358830
x150	(-1)	364674	278720	364674
x151	(-6)	1342	1342	1342
x152	(-5)	1331	1331	2956
x153	(-4)	1319	1319	500
x154	(-3)	1306	1306	500
x155	(-2)	1293	1293	1293
x156	(-1)	1280	1280	1280
x157	(-5)	354725	278658	354725
x158	(-4)	354006	278669	758678
x159	(-3)	351593	278681	150000
x160	(-2)	353079	278694	150000
x161	(-1)	358830	278707	358830
x162	(0)	364674	278720	364674
x163	(-5)	1342	1342	1342
x164	(-4)	1331	1331	2956
x165	(-3)	1319	1319	500
x166	(-2)	1306	1306	500
x167	(-1)	1293	1293	1293
x168	(0)	1280	1280	1280
x169	(-4)	354725	278658	354725
x170	(-3)	354006	278669	758678

x171	(-2)	351593	278681	150000
x172	(-1)	353079	278694	150000
x173	(0)	358830	278707	358830
x174	(1)	364674	278720	364674
x175	(-3)	354725	278658	354725
x176	(-2)	354006	278669	758678
x177	(-1)	351593	278681	150000
x178	(0)	353079	278694	150000
x179	(1)	358830	278707	358830
x180	(2)	364674	278720	364674
x181	(-4)	1342	1342	1342
x182	(-3)	1331	1331	2956
x183	(-2)	1319	1319	500
x184	(-1)	1306	1306	500
x185	(0)	1293	1293	1293
x186	(1)	1280	1280	1280
x187	(-3)	1342	1342	1342
x188	(-2)	1331	1331	2956
x189	(-1)	1319	1319	500
x190	(0)	1306	1306	500
x191	(1)	1293	1293	1293
x192	(2)	1280	1280	1280
x193	(-1)	354725	278658	354725
x194	(0)	354006	278669	758678
x195	(1)	351593	278681	150000
x196	(2)	353079	278694	150000
x197	(3)	358830	278707	358830
x198	(4)	364674	278720	364674
x199	(-1)	1342	1342	1342



x200	(0)	1331	1331	2956
x201	(1)	1319	1319	500
x202	(2)	1306	1306	500
x203	(3)	1293	1293	1293
x204	(4)	1280	1280	1280
x205	(-4)	354725	278658	354725
x206	(-3)	354006	278669	758678
x207	(-2)	351593	278681	150000
x208	(-1)	353079	278694	150000
x209	(0)	358830	278707	358830
x210	(1)	364674	278720	364674
x211	(-3)	354725	278658	354725
x212	(-2)	354006	278669	758678
x213	(-1)	351593	278681	150000
x214	(0)	353079	278694	150000
x215	(1)	358830	278707	358830
x216	(2)	364674	278720	364674
x217	(-4)	1342	1342	1342
x218	(-3)	1331	1331	2956
x219	(-2)	1319	1319	500
x220	(-1)	1306	1306	500
x221	(0)	1293	1293	1293
x222	(1)	1280	1280	1280
x223	(-3)	1342	1342	1342
x224	(-2)	1331	1331	2956
x225	(-1)	1319	1319	500
x226	(0)	1306	1306	500
x227	(1)	1293	1293	1293
x228	(2)	1280	1280	1280

x229	(-1)	354725	278658	354725
x230	(0)	354006	278669	408678
x231	(1)	351593	278681	296921
x232	(2)	353079	278694	353079
x233	(3)	358830	278707	358830
x234	(4)	364674	278720	364674
x235	(-1)	1342	1342	1342
x236	(0)	1331	1331	1331
x237	(1)	1319	1319	1319
x238	(2)	1306	1306	1306
x239	(3)	1293	1293	1293
x240	(4)	1280	1280	1280
x241	(0)	97022	71743	97022
x242	(1)	94295	0	94295
x243	(2)	89822	51892	89822
x244	(3)	89197	0	89197
x245	(4)	92784	219108	92784
x246	(5)	96412	216789	96412
x247	(0)	206915	206915	206915
x248	(1)	206798	174968	206798
x249	(2)	206681	171699	206681
x250	(3)	206564	221376	206564
x251	(4)	206447	0	206447
x252	(5)	206331	0	206331
x253	(0)	50788	0	50788
x254	(1)	52913	103701	52913
x255	(2)	55090	55090	55090
x256	(3)	57318	57318	57318
x257	(4)	59599	59599	59599
x258	(5)	61931	61931	61931

x259	(0)	1106	1106	1106
x260	(1)	1095	1095	1095
x261	(2)	1083	1083	1083
x262	(3)	1070	1070	1070
x263	(4)	1057	1057	1057
x264	(5)	1044	1044	1044
x265	(0)	236	236	236
x266	(1)	236	236	236
x267	(2)	236	236	236
x268	(3)	236	236	236
x269	(4)	236	236	236
x270	(5)	236	236	236
x271	(0)	0	25279	0
x272	(1)	0	119574	0
x273	(2)	0	157504	0
x274	(3)	0	246701	0
x275	(4)	0	120377	0
x276	(5)	0	0	0
x277	(0)	0	0	0
x278	(1)	0	31830	0
x279	(2)	0	66812	0
x280	(3)	0	52000	0
x281	(4)	0	258447	0
x282	(5)	0	464778	0
x283	(0)	0	50788	0
x284	(1)	0	0	0
x285	(2)	0	0	0
x286	(3)	0	0	0
x287	(4)	0	0	0
x288	(5)	0	0	0

x289	(0)	0	0	0
x294	(5)	0	0	0
x295	(0)	0	0	0
x300	(5)	0	0	0

模式包含 6 個時段共有 300 個變數。經最佳化求解，取顧客在  $t = 0$  時下訂單之決策結果如圖 5.26 所示，由決策結果發現以下情形：

- 各節點之存貨量皆為零，物料及零件皆是依下游需求，於所需時段運至零件商或裝配廠於當期生產及裝配，並運送出，因此在無運量限制、無產能限制、無初期存貨，設定最佳狀態下可獲得最大利潤。

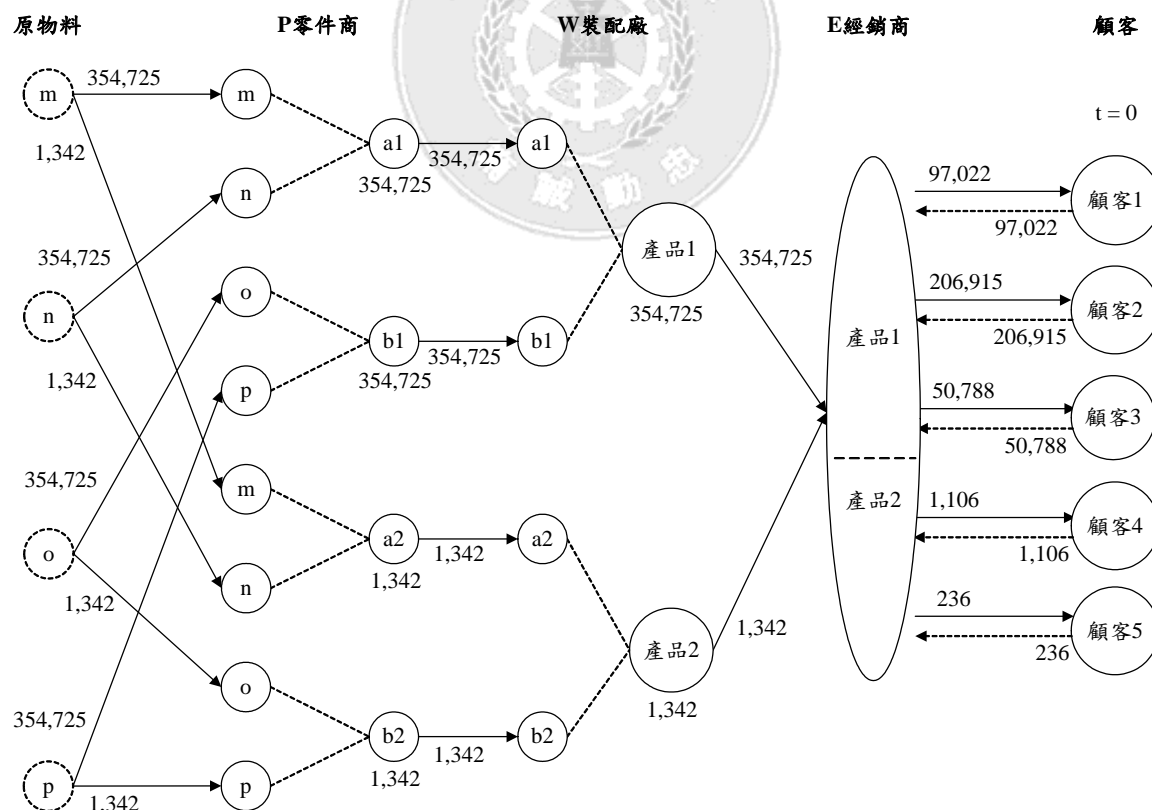


圖 5.26 最佳化結果(訂單在  $t=0$ )

## 敏感度分析

為了解模式之特性，除進行最佳化求解外，並設定兩種情境，分別為物料限制以及產能限制，以觀察最佳化結果的不同。經最佳化求得結果如附錄 3 所示，而各情境所產生之結果說明如下：

### 情境一、物料限制

為檢視物料不足下模式的決策方式，將將物料 m 每一期上限縮小成 280,000，使物料持續不足；經最佳化求解，取顧客在  $t=2$  時下訂單之決策結果如圖 5.27 所示，由決策結果發現以下情形：

- 物料不足的情形下除導致零件生產量減少，因零件所需物料不只一種，因此當一物料不足時相對會影響其他物料，使相關物料運量減少。
- 物料發生不足時將會一連串影響下游零件商、裝配廠及經銷商，運量、產量、銷售量、未結清訂單皆受到影響。
- 此時經銷商需以產品收益為考量，分配產品之銷售數量，由最佳化可以發現，因產品 2 之單價較產品 1 單價為高，因此在物料分配上先滿足產品 2 之需求，再將剩餘資源分配給產品 1；而在產品 1 銷售至不同顧客間，亦依不同出售單價予以分配。

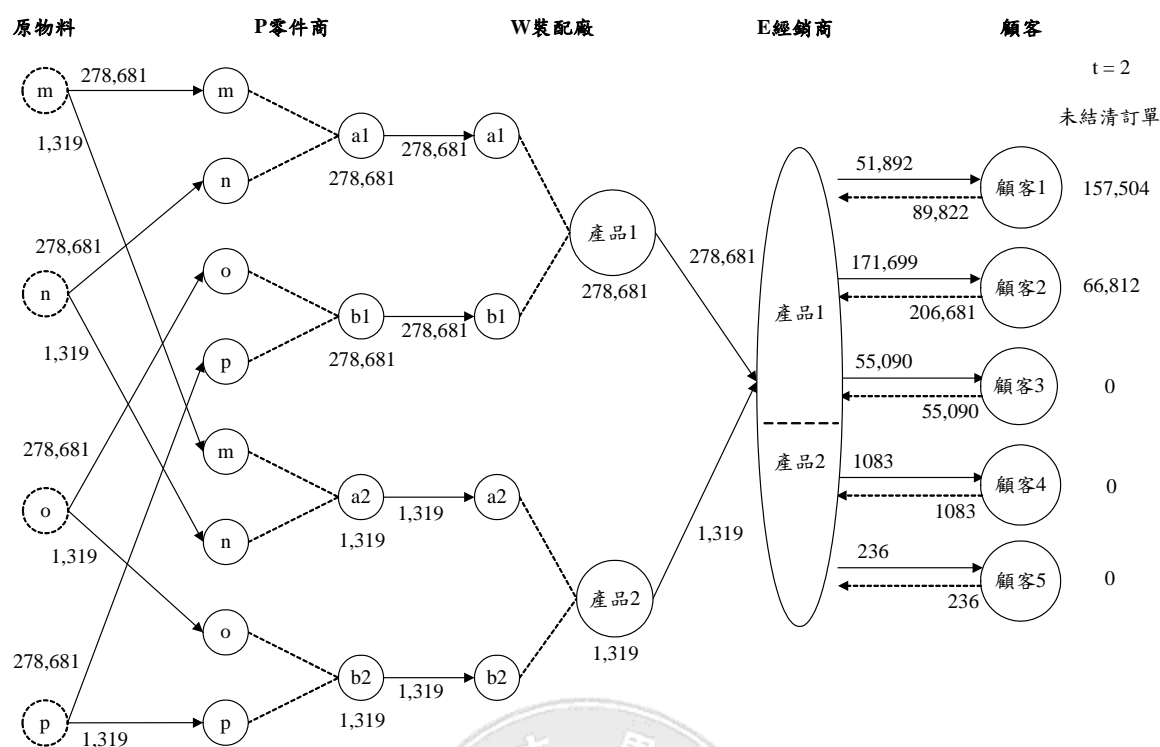


圖 5.27 情境一最佳化結果(訂單在  $t=2$ )

## 情境二、產能限制

為檢視產能不足下模式的決策方式，設計兩裝配廠所生產之兩個產品在  $t=1,2$  發生產能不足；經最佳化求解，取裝配廠在  $t=0$  時下訂單之決策結果如圖 5.28 所示，由決策結果發現以下情形：

- 裝配廠會在前一期增加零件訂單以事先生產並庫存產品，在整體效益的考量下不僅是裝配廠進行存貨動作，經銷商也是事先加大訂貨量預先存貨。
- 因產能不足之情形非整個最佳化時段持續發生，因此在事先生產進行存貨的決策下無未結清訂單發生。
- 在存貨成本設計上，產品在經銷商之單位存貨大於在裝配廠之單位存貨，因此產品 2 是將產品庫存在裝配廠，在經銷商無存貨量；而產品 1 則受限於裝配廠的存貨上限，因此在裝配廠及經銷商均產生存貨。

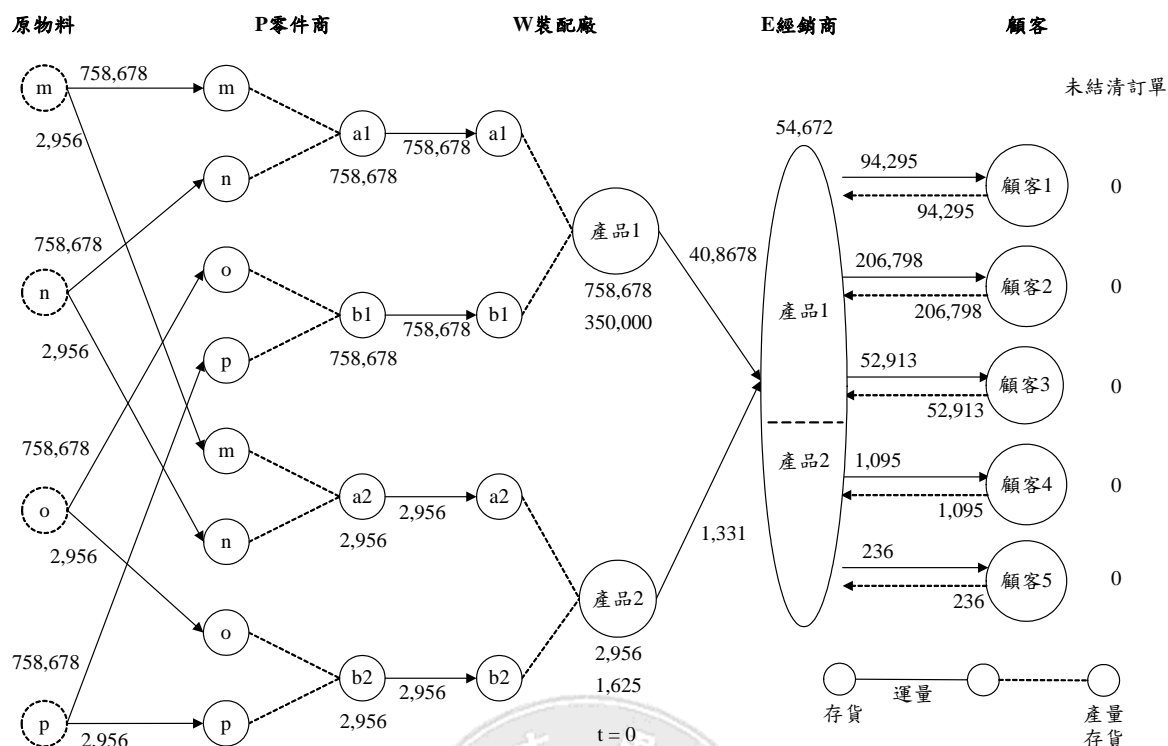


圖 5.28 情境二最佳化結果(裝配廠在  $t=0$ )

### 5.3.3 結果討論

根據需求模式構建及最佳化的數值實驗所產生的結果，歸納出以下幾點：

1. 藉由實際需求資料建構需求預測模式可以發現資料有都其不同之趨勢或週期性，因此以不同筆數之歷史資料確實可更精確的校估出模式參數。
2. 實驗使用 ARIMA 構建需求預測模型，在構建模式的過程中發現 ARIMA 模式確實可藉由差分方式處理不穩定數列，且因其模式是包含 MA 及 AR 模式特性因此在預測上更具彈性，但在預測具季節特性之需求資料時可以發現預測準度較差。
3. 在最佳化求解過程中有先進行一基礎實驗，在基礎實驗中發現如運量上限持續導致無法滿足需求時，此時未結清訂單會出現持續累積情形。
4. 在最佳化求解結果可以發現均無存貨及未結清訂單發生，主要是因

為無運量及產量限制，物料量亦足以供應全體路網所需，因此供應鏈在營運上可以達到豐田生產方式中及時化的概念，依所需下游產品需求量決定上游運量及產量。

5. 整體供應鏈路網受許多因素影響，如經由敏感度分析可以發現，在物料不足持續不足的情形下，下游零件商、裝配廠及經銷商皆受影響，在確定無法滿足需求的情形下僅能先滿足利潤較高之產品；如事件非持續發生，如預設僅幾個時段之產能不足，則可由受影響之供應鏈階層調整訂貨量，分擔存貨成本。





## 第六章 結論與建議

### 6.1 結論

本研究利用模式預測控制之概念，建構一供應鏈模型以描述基礎供應鏈之營運情形，模型中包含工廠、倉庫、配送中心與零售商之間的相互關係，並校估出一需求預測模式以預測未來之需求供最佳化過程產生決策。實驗中設計不同情境以產生決策結果，檢視模式。

研究中之需求預測模型是利用 SPSS 統計軟體產生 ARIMA 模型以進行需求預測，接著利用 ILOG-CPLEX 求解供應鏈模式，獲得各點間在各時段下之運量決策。研究中提出以下結論：

1. 本研究構建一基礎供應鏈模式，依據需求預測結果進行最佳化，可提供整體運量決策。
2. 研究中所構建之模式是以自行車產業上中下游密切之關係為基礎進行案例設計，上游可以針對下游需求量決定產能及運量，產生各階層存貨量皆為零的情形。
3. 架構中決策變數為運量決策，設計存貨及未結清訂單為控制項目，但在決策方式資訊不足情形下，在無事件發生情形時為零存貨且無未結清訂單，因此無法進行控制行為。
4. 模式預測控制在模式建構上，如模式與系統真實運作情況越符合，則所獲得結果會越準確。在研究中主要是以模式預測控制之概念為架構應用在供應鏈管理上，建構需求預測模式及供應鏈模式，因此需求模式若能越準確、供應鏈模式若與真實供應鏈營運狀況越切合，則所預測之決策為供應鏈管理所能帶來更大的效益。
5. 於數值實驗中發現，ARIMA 模式確實可捕捉需求時間序列之趨勢；再應用滾動平面法之概念，可針對需求資料之不同特性，校估出適合用來推估需求預測模式之歷史資料筆數。

6. 在敏感度分析中發現在物料不足或是產能不足的事件時，對其上下游皆會產生影響，即必須先進行存貨動作以防止缺貨發生；而事件發生的期間對供應鏈影響亦很大，如事件持續發生勢必會產生缺貨及未結清訂單。

## 6.2 建議

本研究提出供應鏈模型及需求預測模型，並進行最佳化求解獲得決策結果。然而其中仍存在著許多課題尚待探討，因此，對於後續研究方向提出以下建議：

1. 研究中已考量產量、運量及存貨，建立供應鏈模型，並藉由模式預測控制分析出決策與控制變數，未來可依據產業之實際需求或限制，設計存貨管理、訂單決策加入模式中，更有效進行控制動作產生決策結果。
2. 在各階層間關係密切的情形下，一但中間有一階層發生無法及時供貨，下游勢必受到影響，因此如何迅速尋找可替代上游供應商十分重要；而在產業朝高單價產品開發的趨勢上，其零件商亦須隨之變動，因此，上游供應商須了解下游需求，隨時依據需求在產品研發及產量上做調整，以能正確且及時供應產品。

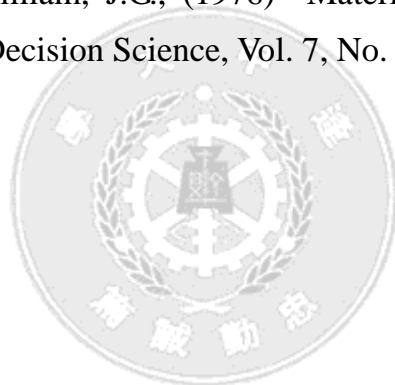
## 參考文獻

1. 仇士元 (2004), 「加總模型的預測效果—以台灣地區汽車銷售量資料為例」, 長庚大學企業管理研究所碩士論文。
2. 王信傑 (2002), 「在供需不確定性因素下之製造商作業規劃模式」, 淡江大學資訊管理學系碩士論文。
3. 交通部 (2004), 「區域級智慧型運輸系統示範計劃—核心交通分析與預測系統」。
4. 郭怡姝 (2001), 「產業之不確定因素對供應鏈管理策略與其績效影響之研究」, 淡江大學資訊管理學系碩士論文。
5. 劉權輝 (2000), 「類神經網路於非線性程序系統控制之應用」, 大業大學食品工程研究所碩士論文。
6. 陳正昌、程炳林(1994), SPSS、SAS、BMDP 統計軟體在多變量統計上的應用, 五南圖書出版公司。
7. 潘曉葦 (2000), 「需求預測模式之建立」, 國立台灣科技大學資訊管理系碩士論文。
8. 蔡柏澧 (2003), 「類神經網路模式預測控制」, 國立清華大學化學工程研究所博士論文。
9. 羅國正 (2000), 「推行供應鏈管理之不確定性因素及其因應策略之研究—以台灣資訊電子業為例」, 國立政治大學資訊管理學系碩士班碩士論文。
10. Backx, T.O., Bosgra, O., and Marquardt, W. (1998) “Towards intentional dynamics in supply chain conscious process operations,” Third International Conference on Foundations of Computer-Aided Process Operations , American Institute of Chemical Engineers.
11. Bose, S., and Pekny, J. F. (2000) “A model predictive framework for planning and scheduling problems: a case study of consumer goods supply chain,” Computers and Chemical Engineering, 24, pp.329-335.
12. Box, G. E. P., Jenkins, G. M. and Reinsel, G. C. (1994) Time Series Analysis: Forecasting and Control, 3rd edition, Prentice Hall International Editors.
13. Braun, M.W., Rivera, D.E., Flores, M.E., Carlyle, W.M., and Kempf, K.G., (2003) “A Model Predictive Control framework for robust management of multi-product, multi-echelon demand networks,” Annual Reviews in

- Control, 27, pp.229-245.
14. Cetinkaya, S., and Lee, C. (2000) "Stock replenishment and shipment scheduling for vendor managed inventory systems," *Management Science*, 46(2), pp.217-232.
  15. Chopra, S., and Meindl, P. (2001) *Supply Chain Management : Strategy, Planning, and Operation*, Prentice Hall INC.
  16. Davis, T., "Effective Supply chain management," *Solan Management Review*, 1993, pp.35-46.
  17. Dejonckheere, J., Disney, S.M., Lambrecht, M.R., and Towill, D.R. (2004) "The impact of information enrichment on the Bullwhip effect in supply chains: A control engineering perspective," *European Journal of Operational Research*, 153, pp.727-750.
  18. Ettl, M., Feigin, G. E., Lin, G., and Yao, D. D. (2000) "A supply network model with base-stock control and service requirements," *Operations Research*, 48(2), pp.216-232.
  19. Forrester, J.W., (1961), *Industrial Dynamics*. MIT Press, Cambridge, MA.
  20. Houlihan, J. B., (1985) "International Supply Chain Management," *International Journal of Physical Distribution and Materials Management*, 15(1), pp.22-38.
  21. Lee, H.L. & Billington, C., (1993) "Material management in de-centralized supply chains , " *Operations Research*, Vol. 41, No. 5, pp.835-847.
  22. Perea, E., Grossmann, I., and Ydstie, E. (2000) "Dynamic modeling and decentralized control of supply chains," *Industrial Engineering Chemistry Research*, 40 (15), pp.3369-3383.
  23. Perea-Lo'pez, E., Ydstie, B. E., and Grossmann, I.E. (2003) "A model predictive control strategy for supply chain optimization," *Computers and Chemical Engineering*, 27, pp.1201-1218.
  24. Sakawa, M., Nishizaki, I., and Uemure, Y. (2001) "Fuzzy programming and profit and cost allocation for a multiproduct and transportation problem," *European Journal of Operational Research*, 131, pp.1-15.
  25. Seferlis, P., and Giannelos, N.F. (2004) "A two-layered optimisation-based control strategy for multi-echelon supply chain networks," *Computers and*

Chemical Engineering 28, pp.799-809.

26. Shah, N. (1998) "Single and multisite planning and scheduling: current status and future challenges," American Institute of Chemical Engineers Symposium Series, pp.75-90.
27. Sivakumar, A. I. (2001) "Multiobjective dynamic scheduling using discrete event simulation," International Journal of Computer Integrated Manufacturing, 14, pp.154-167.
28. Towill, D. R. (1982) "Dynamic analysis of an inventory and order based production control system," International Journal of Product Research, 20, pp.671-687.
29. Towill, D. R. (1996) "Industrial dynamics modeling of supply chains," Logistics Information Management, 9, pp.43-56.
30. Whybark, D.C & William, J.G., (1976) "Material Requirement Planning under Uncertainty," Decision Science, Vol. 7, No. 4, pp.595-606.



## 附錄一 自行車出口數量及出口單價資料

### 自行車在北美自由貿易區(NAFTA)出口數量

	車輛數		車輛數		車輛數		車輛數		車輛數
1	176,829	13	154,551	25	115,706	37	84,385	49	89,722
2	219,360	14	141,775	26	67,926	38	86,240	50	80,075
3	256,531	15	104,431	27	102,668	39	76,592	51	72,955
4	302,121	16	59,067	28	108,661	40	81,934	52	74,634
5	228,635	17	77,653	29	110,236	41	78,091	53	69,300
6	282,151	18	99,426	30	117,033	42	70,579	54	107,080
7	272,373	19	214,958	31	111,243	43	83,822	55	109,948
8	199,763	20	123,284	32	90,394	44	100,297	56	84,759
9	223,589	21	115,952	33	117,273	45	108,511	57	81,258
10	223,535	22	105,027	34	87,868	46	82,156	58	86,639
11	113,604	23	90,675	35	61,056	47	59,306	59	80,546
12	130,122	24	97,717	36	63,757	48	67,393	60	78,196

### 自行車在北美自由貿易區(NAFTA)出口單價(US\$)

	單價		單價		單價		單價		單價
1	110.15	13	115.54	25	195.07	37	175.85	49	251.91
2	105.89	14	139.42	26	116.51	38	122.23	50	203.4
3	93.78	15	134.05	27	106.03	39	144.33	51	173.13
4	91.82	16	129.95	28	124.66	40	143.57	52	152.94
5	85.48	17	96.76	29	126.34	41	184.68	53	182.18
6	86.8	18	111.33	30	160.71	42	200.55	54	168.35
7	89.77	19	70.1	31	172.12	43	194.36	55	193.04
8	112.35	20	114.41	32	156.68	44	174.26	56	260.81
9	108.17	21	137.87	33	224.2	45	222.01	57	285.18
10	134.92	22	168.5	34	236.23	46	223.69	58	324.96
11	175.38	23	191.42	35	268.11	47	244.59	59	292.67
12	141.49	24	168.25	36	248.29	48	254.92	60	251.91

### 自行車在歐盟(EU)出口數量

	車輛數		車輛數		車輛數		車輛數		車輛數
1	264,089	13	209,737	25	207,069	37	213,056	49	240,753
2	297,692	14	138,875	26	153,170	38	189,766	50	198,155

3	190,486	15	91,486	27	102,792	39	113,822	51	180,963
4	101,755	16	89,647	28	127,789	40	90,667	52	157,240
5	130,967	17	76,361	29	111,900	41	148,745	53	131,610
6	153,915	18	88,167	30	147,909	42	133,223	54	203,662
7	172,563	19	118,553	31	147,377	43	190,801	55	252,592
8	172,552	20	167,572	32	171,713	44	193,115	56	254,579
9	173,447	21	103,327	33	121,660	45	150,260	57	211,913
10	170,755	22	6,701	34	143,574	46	179,593	58	258,295
11	221,451	23	142,410	35	168,336	47	169,926	59	303,027
12	179,168	24	121,037	36	179,694	48	212,937	60	277,387

自行車在歐盟(EU)出口單價(US\$)

	單價		單價		單價		單價		單價
1	141.14	13	131.97	25	129.76	37	142.54	49	149.11
2	147.9	14	135.16	26	109.74	38	125.93	50	136.04
3	141.19	15	117.85	27	105.02	39	100.46	51	122.18
4	134.46	16	99.63	28	80.3	40	101.64	52	112.47
5	105.27	17	69.6	29	84.27	41	99.94	53	116.84
6	98.91	18	100.69	30	91.99	42	128.42	54	128.71
7	117.76	19	112.83	31	121.03	43	129.82	55	128.11
8	119.11	20	103.99	32	127.12	44	130.87	56	153.39
9	126.02	21	106.11	33	153.77	45	162.15	57	176.62
10	163.48	22	153.65	34	149.89	46	175.26	58	176.53
11	143.66	23	143.53	35	141.46	47	153.04	59	160.53
12	127.46	24	127.63	36	144.06	48	144.49	60	145.74

自行車在歐協(EFTA)出口數量

	車輛數		車輛數		車輛數		車輛數		車輛數
1	18,306	13	41,331	25	24,878	37	33,852	49	19,228
2	38,999	14	29,504	26	10,253	38	16,154	50	18,963
3	11,700	15	17,711	27	3,112	39	8,811	51	4,804
4	14,575	16	9,567	28	5,165	40	5,150	52	4,668
5	4,993	17	543	29	2,341	41	4,781	53	2,916
6	3,626	18	1,684	30	3,458	42	2,427	54	4,348
7	3,586	19	2,286	31	3,145	43	4,555	55	4,072
8	8,189	20	3,503	32	3,258	44	6,800	56	10,124

9	15,630	21	6,689	33	8,344	45	6,666	57	11,867
10	15,619	22	10,158	34	13,863	46	14,073	58	19,695
11	27,785	23	27,479	35	19,795	47	22,749	59	27,343
12	44,661	24	22,705	36	27,383	48	27,890	60	18,651

自行車在歐協(EFTA)出口單價(US\$)

	單價		單價		單價		單價		單價
1	195.52	13	152.98	25	155.87	37	177.43	49	238.56
2	195.52	14	149.81	26	140.76	38	171.25	50	251.59
3	176.16	15	153.14	27	176.03	39	199.75	51	280.73
4	250.79	16	198.38	28	127.72	40	203.63	52	182.3
5	130.74	17	173.81	29	144.63	41	210.03	53	209.5
6	208.59	18	129.59	30	176.81	42	208.74	54	190.93
7	154.46	19	146.01	31	159.45	43	179.04	55	247.99
8	203.9	20	111.22	32	274	44	284.2	56	186.62
9	160.45	21	172.64	33	254.76	45	222.44	57	301.85
10	175.24	22	199.69	34	177.17	46	245.6	58	256.83
11	252.29	23	137.45	35	185.56	47	197.19	59	216.03
12	178.37	24	166.47	36	180.47	48	180.47	60	244.77

電動自行車在北美自由貿易區(NAFTA)出口數量

	車輛數		車輛數		車輛數		車輛數		車輛數
1	1,112	13	500	25	522	37	4,954	49	198
2	303	14	1,060	26	1,874	38	977	50	354
3	1,180	15	490	27	1,594	39	569	51	167
4	1,922	16	121	28	1,347	40	373	52	493
5	2,007	17	593	29	2,971	41	59	53	155
6	4,223	18	2,530	30	1,051	42	237	54	233
7	2,630	19	1,189	31	4,873	43	15	55	335
8	3,085	20	1,847	32	11,950	44	39	56	587
9	3,266	21	419	33	6,879	45	460	57	433
10	2,374	22	453	34	3,859	46	238	58	452
11	1,212	23	497	35	886	47	155	59	466
12	912	24	512	36	735	48	185	60	155



電動自行車在北美自由貿易區(NAFTA)出口單價(US\$)

	單價		單價		單價		單價		單價
1	579.13	13	284.38	25	210.78	37	118.05	49	366.77
2	578.85	14	245.25	26	249.24	38	239	50	601.75
3	657.04	15	378.66	27	313.89	39	514	51	641.6
4	310.04	16	399.7	28	358.06	40	451.68	52	223.86
5	451.07	17	179.98	29	246.04	41	532.1	53	539.07
6	341.35	18	337	30	258.71	42	330.67	54	560.7
7	332.16	19	374.22	31	238.27	43	285.53	55	385.56
8	414.38	20	410.45	32	153.02	44	531.97	56	62.59
9	421.69	21	343.17	33	164.52	45	323.69	57	396.55
10	336.95	22	276.2	34	226.98	46	513.91	58	534
11	581.47	23	236.22	35	197.96	47	504.42	59	552.73
12	392.82	24	245.7	36	251.22	48	456	60	591.7

電動自行車在歐盟(EU)出口數量

	車輛數		車輛數		車輛數		車輛數		車輛數
1	366	13	1,266	25	56	37	183	49	202
2	719	14	1,398	26	100	38	65	50	151
3	590	15	601	27	106	39	19	51	39
4	1,412	16	19	28	696	40	103	52	23
5	314	17	488	29	309	41	200	53	10
6	300	18	276	30	120	42	253	54	203
7	603	19	241	31	265	43	4	55	125
8	330	20	43	32	194	44	154	56	70
9	1,612	21	51	33	105	45	497	57	78
10	1,220	22	233	34	10	46	602	58	61
11	1,278	23	432	35	23	47	580	59	174
12	458	24	7	36	395	48	600	60	119

電動自行車在歐盟(EU)出口單價(US\$)

	單價		單價		單價		單價		單價
1	340.8	13	408.77	25	345.8	37	334.94	49	541.12
2	461.14	14	355.59	26	287.76	38	261.8	50	699.74
3	463.25	15	390.93	27	366.83	39	717.89	51	488.18

4	434	16	282.16	28	231.19	40	182.9	52	432
5	310.4	17	414.69	29	301.27	41	513.53	53	377
6	299.92	18	532.69	30	241.07	42	408.38	54	491.02
7	352.34	19	281.79	31	240.28	43	425.25	55	982.53
8	331.72	20	249.33	32	106.22	44	319.86	56	580.54
9	339.28	21	511.53	33	325.7	45	290.81	57	193.37
10	343.9	22	345.76	34	181	46	305.27	58	304.67
11	374.73	23	271.83	35	408.96	47	432	59	582.25
12	374.34	24	379.43	36	423.04	48	480.7	60	612.7



## 附錄二 多階層多產品最佳化結果

			情境一	情境二
無運量限制、 無初期存貨			物料不足	產能不足
變數	時段		$X_{1.2.3.4.5.6} < 280,000$	$X_{194.195} < 150,000$ $X_{200.201} < 500$
		求解結果	求解結果	求解結果
x1	(-5)	0	0	0
x2	(-4)	0	0	0
x3	(-3)	0	0	0
x4	(-2)	0	0	0
x5	(-1)	0	0	0
x6	(0)	0	0	0
x7	(-5)	0	0	0
x8	(-4)	0	0	0
x9	(-3)	0	0	0
x10	(-2)	0	0	0
x11	(-1)	0	0	0
x12	(0)	0	0	0
x13	(-4)	0	0	0
x14	(-3)	0	0	0
x15	(-2)	0	0	0
x16	(-1)	0	0	0
x17	(0)	0	0	0
x18	(1)	0	0	0
x19	(-4)	0	0	0
x20	(-3)	0	0	0
x21	(-2)	0	0	0
x22	(-1)	0	0	0
x23	(0)	0	0	0
x24	(1)	0	0	0
x25	(-4)	0	0	0
x26	(-3)	0	0	0
x27	(-2)	0	0	0
x28	(-1)	0	0	0
x29	(0)	0	0	0

x30	(1)	0	0	0
x31	(-3)	0	0	0
x32	(-2)	0	0	0
x33	(-1)	0	0	0
x34	(0)	0	0	0
x35	(1)	0	0	0
x36	(2)	0	0	0
x37	(-5)	0	0	0
x38	(-4)	0	0	0
x39	(-3)	0	0	0
x40	(-2)	0	0	0
x41	(-1)	0	0	0
x42	(0)	0	0	0
x43	(-5)	0	0	0
x44	(-4)	0	0	0
x45	(-3)	0	0	0
x46	(-2)	0	0	0
x47	(-1)	0	0	0
x48	(0)	0	0	0
x49	(-4)	0	0	0
x50	(-3)	0	0	0
x51	(-2)	0	0	0
x52	(-1)	0	0	0
x53	(0)	0	0	0
x54	(1)	0	0	0
x55	(-4)	0	0	0
x56	(-3)	0	0	0
x57	(-2)	0	0	0
x58	(-1)	0	0	0
x59	(0)	0	0	0
x60	(1)	0	0	0
x61	(-4)	0	0	0
x62	(-3)	0	0	0
x63	(-2)	0	0	0
x64	(-1)	0	0	0
x65	(0)	0	0	0
x66	(1)	0	0	0
x67	(-3)	0	0	0

x68	(-2)	0	0	0
x69	(-1)	0	0	0
x70	(0)	0	0	0
x71	(1)	0	0	0
x72	(2)	0	0	0
x73	(-2)	0	0	0
x74	(-1)	0	0	0
x75	(0)	0	0	0
x76	(1)	0	0	0
x77	(2)	0	0	0
x78	(3)	0	0	0
x79	(-2)	0	0	0
x80	(-1)	0	0	0
x81	(0)	0	0	0
x82	(1)	0	0	0
x83	(2)	0	0	0
x84	(3)	0	0	0
x85	(-1)	0	0	0
x86	(0)	0	0	350000
x87	(1)	0	0	203079
x88	(2)	0	0	0
x89	(3)	0	0	0
x90	(4)	0	0	0
x91	(-2)	0	0	0
x92	(-1)	0	0	0
x93	(0)	0	0	0
x94	(1)	0	0	0
x95	(2)	0	0	0
x96	(3)	0	0	0
x97	(-2)	0	0	0
x98	(-1)	0	0	0
x99	(0)	0	0	0
x100	(1)	0	0	0
x101	(2)	0	0	0
x102	(3)	0	0	0
x103	(-1)	0	0	0
x104	(0)	0	0	1625

x105	(1)	0	0	806
x106	(2)	0	0	0
x107	(3)	0	0	0
x108	(4)	0	0	0
x109	(0)	0	0	0
x110	(1)	0	0	54672
x111	(2)	0	0	0
x112	(3)	0	0	0
x113	(4)	0	0	0
x114	(5)	0	0	0
x115	(0)	0	0	0
x116	(1)	0	0	0
x117	(2)	0	0	0
x118	(3)	0	0	0
x119	(4)	0	0	0
x120	(5)	0	0	0
x121	(-7)	354725	278658	354725
x122	(-6)	354006	278669	758678
x123	(-5)	351593	278681	150000
x124	(-4)	353079	278694	150000
x125	(-3)	358830	278707	358830
x126	(-2)	364674	278720	364674
x127	(-7)	1342	1342	1342
x128	(-6)	1331	1331	2956
x129	(-5)	1319	1319	500
x130	(-4)	1306	1306	500
x131	(-3)	1293	1293	1293
x132	(-2)	1280	1280	1280
x133	(-6)	354725	278658	354725
x134	(-5)	354006	278669	758678
x135	(-4)	351593	278681	150000
x136	(-3)	353079	278694	150000
x137	(-2)	358830	278707	358830
x138	(-1)	364674	278720	364674
x139	(-6)	1342	1342	1342
x140	(-5)	1331	1331	2956
x141	(-4)	1319	1319	500

x142	(-3)	1306	1306	500
x143	(-2)	1293	1293	1293
x144	(-1)	1280	1280	1280
x145	(-6)	354725	278658	354725
x146	(-5)	354006	278669	758678
x147	(-4)	351593	278681	150000
x148	(-3)	353079	278694	150000
x149	(-2)	358830	278707	358830
x150	(-1)	364674	278720	364674
x151	(-6)	1342	1342	1342
x152	(-5)	1331	1331	2956
x153	(-4)	1319	1319	500
x154	(-3)	1306	1306	500
x155	(-2)	1293	1293	1293
x156	(-1)	1280	1280	1280
x157	(-5)	354725	278658	354725
x158	(-4)	354006	278669	758678
x159	(-3)	351593	278681	150000
x160	(-2)	353079	278694	150000
x161	(-1)	358830	278707	358830
x162	(0)	364674	278720	364674
x163	(-5)	1342	1342	1342
x164	(-4)	1331	1331	2956
x165	(-3)	1319	1319	500
x166	(-2)	1306	1306	500
x167	(-1)	1293	1293	1293
x168	(0)	1280	1280	1280
x169	(-4)	354725	278658	354725
x170	(-3)	354006	278669	758678
x171	(-2)	351593	278681	150000
x172	(-1)	353079	278694	150000
x173	(0)	358830	278707	358830
x174	(1)	364674	278720	364674
x175	(-3)	354725	278658	354725
x176	(-2)	354006	278669	758678
x177	(-1)	351593	278681	150000
x178	(0)	353079	278694	150000

x179	(1)	358830	278707	358830
x180	(2)	364674	278720	364674
x181	(-4)	1342	1342	1342
x182	(-3)	1331	1331	2956
x183	(-2)	1319	1319	500
x184	(-1)	1306	1306	500
x185	(0)	1293	1293	1293
x186	(1)	1280	1280	1280
x187	(-3)	1342	1342	1342
x188	(-2)	1331	1331	2956
x189	(-1)	1319	1319	500
x190	(0)	1306	1306	500
x191	(1)	1293	1293	1293
x192	(2)	1280	1280	1280
x193	(-1)	354725	278658	354725
x194	(0)	354006	278669	758678
x195	(1)	351593	278681	150000
x196	(2)	353079	278694	150000
x197	(3)	358830	278707	358830
x198	(4)	364674	278720	364674
x199	(-1)	1342	1342	1342
x200	(0)	1331	1331	2956
x201	(1)	1319	1319	500
x202	(2)	1306	1306	500
x203	(3)	1293	1293	1293
x204	(4)	1280	1280	1280
x205	(-4)	354725	278658	354725
x206	(-3)	354006	278669	758678
x207	(-2)	351593	278681	150000
x208	(-1)	353079	278694	150000
x209	(0)	358830	278707	358830
x210	(1)	364674	278720	364674
x211	(-3)	354725	278658	354725
x212	(-2)	354006	278669	758678
x213	(-1)	351593	278681	150000
x214	(0)	353079	278694	150000
x215	(1)	358830	278707	358830



x216	(2)	364674	278720	364674
x217	(-4)	1342	1342	1342
x218	(-3)	1331	1331	2956
x219	(-2)	1319	1319	500
x220	(-1)	1306	1306	500
x221	(0)	1293	1293	1293
x222	(1)	1280	1280	1280
x223	(-3)	1342	1342	1342
x224	(-2)	1331	1331	2956
x225	(-1)	1319	1319	500
x226	(0)	1306	1306	500
x227	(1)	1293	1293	1293
x228	(2)	1280	1280	1280
x229	(-1)	354725	278658	354725
x230	(0)	354006	278669	408678
x231	(1)	351593	278681	296921
x232	(2)	353079	278694	353079
x233	(3)	358830	278707	358830
x234	(4)	364674	278720	364674
x235	(-1)	1342	1342	1342
x236	(0)	1331	1331	1331
x237	(1)	1319	1319	1319
x238	(2)	1306	1306	1306
x239	(3)	1293	1293	1293
x240	(4)	1280	1280	1280
x241	(0)	97022	71743	97022
x242	(1)	94295	0	94295
x243	(2)	89822	51892	89822
x244	(3)	89197	0	89197
x245	(4)	92784	219108	92784
x246	(5)	96412	216789	96412
x247	(0)	206915	206915	206915
x248	(1)	206798	174968	206798
x249	(2)	206681	171699	206681
x250	(3)	206564	221376	206564
x251	(4)	206447	0	206447
x252	(5)	206331	0	206331

x253	(0)	50788	0	50788
x254	(1)	52913	103701	52913
x255	(2)	55090	55090	55090
x256	(3)	57318	57318	57318
x257	(4)	59599	59599	59599
x258	(5)	61931	61931	61931
x259	(0)	1106	1106	1106
x260	(1)	1095	1095	1095
x261	(2)	1083	1083	1083
x262	(3)	1070	1070	1070
x263	(4)	1057	1057	1057
x264	(5)	1044	1044	1044
x265	(0)	236	236	236
x266	(1)	236	236	236
x267	(2)	236	236	236
x268	(3)	236	236	236
x269	(4)	236	236	236
x270	(5)	236	236	236
x271	(0)	0	25279	0
x272	(1)	0	119574	0
x273	(2)	0	157504	0
x274	(3)	0	246701	0
x275	(4)	0	120377	0
x276	(5)	0	0	0
x277	(0)	0	0	0
x278	(1)	0	31830	0
x279	(2)	0	66812	0
x280	(3)	0	52000	0
x281	(4)	0	258447	0
x282	(5)	0	464778	0
x283	(0)	0	50788	0
x284	(1)	0	0	0
x285	(2)	0	0	0
x286	(3)	0	0	0
x287	(4)	0	0	0
x288	(5)	0	0	0
x289	(0)	0	0	0

供應鏈管理下需求預測控制之研究

x290	(1)	0	0	0
x291	(2)	0	0	0
x292	(3)	0	0	0
x293	(4)	0	0	0
x294	(5)	0	0	0
x295	(0)	0	0	0
x296	(1)	0	0	0
x297	(2)	0	0	0
x298	(3)	0	0	0
x299	(4)	0	0	0
x300	(5)	0	0	0



### 附錄三 應用CPLEX求解模式程式

```
/* mipex1.c - Entering and optimizing a MIP problem */

/* Bring in the CPLEX function declarations and the C library
   header file stdio.h with the include of cplex.h. */

#include <ilcplex/cplex.h>
#include <stdlib.h>

/* Bring in the declarations for the string functions */

#include <string.h>

/* Include declaration for function at end of program */

#ifndef CPX_PROTOTYPE_MIN

static int
    setproblemdata (char **probname_p, int *numcols_p, int *numrows_p,
                    int *objsen_p, double **obj_p, double **rhs_p,
                    char **sense_p, int **matbeg_p, int **matcnt_p,
                    int **matind_p, double **matval_p,
                    double **lb_p, double **ub_p, char **ctype_p);

static void
    free_and_null (char **ptr);

#else

static int
    setproblemdata ();

static void
    free_and_null ();

#endif

#define NUMROWS    366
#define NUMCOLS    300
#define NUMNZ      1108

#ifndef CPX_PROTOTYPE_MIN
int
main (void)
#else
int
main ()
#endif
{
/* Declare pointers for the variables and arrays that will contain
```

the data which define the LP problem. The setproblemdata() routine allocates space for the problem data. \*/

```
char    *probname = NULL;
int      numcols;
int      numrows;
int      objsen;
double   *obj = NULL;
double   *rhs = NULL;
char     *sense = NULL;
int      *matbeg = NULL;
int      *matcnt = NULL;
int      *matind = NULL;
double   *matval = NULL;
double   *lb = NULL;
double   *ub = NULL;
char     *ctype = NULL;
```

/\* Declare and allocate space for the variables and arrays where we will store the optimization results including the status, objective value, variable values, and row slacks. \*/

```
int      solstat;
double   objval;
double   x[NUMCOLS];
double   slack[NUMROWS];
```

```
CPXENVptr env = NULL;
CPXLPptr  lp = NULL;
int       status;
int       i, j;
int       cur_numrows, cur_numcols;
FILE *fptr;
char fname[]="d://result.txt";
fptr=fopen(fname,"a");
```

/\* Initialize the CPLEX environment \*/

```
env = CPXopenCPLEX (&status);
```

/\* If an error occurs, the status value indicates the reason for failure. A call to CPXgeterrorstring will produce the text of the error message. Note that CPXopenCPLEX produces no output, so the only way to see the cause of the error is to use CPXgeterrorstring. For other CPLEX routines, the errors will be seen if the CPX\_PARAM\_SCRIND indicator is set to CPX\_ON. \*/

```
if ( env == NULL ) {
    char  errmsg[1024];
    fprintf (stderr, "Could not open CPLEX environment.\n");
    CPXgeterrorstring (env, status, errmsg);
    fprintf (stderr, "%s", errmsg);
}
```



```
        goto TERMINATE;
    }

    /* Turn on output to the screen */

    status = CPXsetintparam (env, CPX_PARAM_SCRIND, CPX_ON);
    if ( status ) {
        fprintf (stderr,
            "Failure to turn on screen indicator, error %d.\n", status);
        goto TERMINATE;
    }

    /* Fill in the data for the problem.  */

    status = setproblemdata (&probname, &numcols, &numrows, &objsen, &obj,
                            &rhs, &sense, &matbeg, &matcnt, &matind, &matval,
                            &lb, &ub, &ctype);

    if ( status ) {
        fprintf (stderr, "Failed to build problem data arrays.\n");
        goto TERMINATE;
    }

    /* Create the problem.  */

    lp = CPXcreateprob (env, &status, probname);

    /* A returned pointer of NULL may mean that not enough memory
       was available or there was some other problem.  In the case of
       failure, an error message will have been written to the error
       channel from inside CPLEX.  In this example, the setting of
       the parameter CPX_PARAM_SCRIND causes the error message to
       appear on stdout.  */

    if ( lp == NULL ) {
        fprintf (stderr, "Failed to create LP.\n");
        goto TERMINATE;
    }

    /* Now copy the problem data into the lp */

    status = CPXcopylp (env, lp, numcols, numrows, objsen, obj, rhs,
                       sense, matbeg, matcnt, matind, matval,
                       lb, ub, NULL);

    if ( status ) {
        fprintf (stderr, "Failed to copy problem data.\n");
        goto TERMINATE;
    }

    /* Now copy the ctype array */

    status = CPXcopyctype (env, lp, ctype);
    if ( status ) {
```

```
    fprintf(stderr, "Failed to copy ctype\n");
    goto TERMINATE;
}

/* Optimize the problem and obtain solution. */

status = CPXmipopt (env, lp);
if ( status ) {
    fprintf(stderr, "Failed to optimize MIP.\n");
    goto TERMINATE;
}

solstat = CPXgetstat (env, lp);

/* Write the output to the screen. */

//output

fprintf(fp, "%d\n", solstat);

status = CPXgetmipobjval (env, lp, &objval);
if ( status ) {
    fprintf(stderr, "No MIP objective value available. Exiting...\n");
    goto TERMINATE;
}

//output

fprintf(fp, "%d\n", objval);

/* The size of the problem should be obtained by asking CPLEX what
   the actual size is, rather than using what was passed to CPXcopylp.
   cur_numrows and cur_numcols store the current number of rows and
   columns, respectively. */

cur_numrows = CPXgetnumrows (env, lp);
cur_numcols = CPXgetnumcols (env, lp);

status = CPXgetmipx (env, lp, x, 0, cur_numcols-1);
if ( status ) {
    fprintf(stderr, "Failed to get optimal integer x.\n");
    goto TERMINATE;
}

status = CPXgetmipslack (env, lp, slack, 0, cur_numrows-1);
if ( status ) {
    fprintf(stderr, "Failed to get optimal slack values.\n");
    goto TERMINATE;
}
```



```
//output
for (i = 0; i < cur_numrows; i++) {
    fprintf(fp, "Row %d: Slack = %10f\n", i, slack[i]);
}

// output
for (j = 0; j < cur_numcols; j++) {

fprintf(fp, "Column %d: Value = %10f\n", j, x[j]);

}

/* Finally, write a copy of the problem to a file. */

status = CPXwriteprob (env, lp, "mipex1.lp", NULL);
if ( status ) {
    fprintf (stderr, "Failed to write LP to disk.\n");
    goto TERMINATE;
}

TERMINATE:

/* Free up the problem as allocated by CPXcreateprob, if necessary */

if ( lp != NULL ) {
    status = CPXfreeprob (env, &lp);
    if ( status ) {
        fprintf (stderr, "CPXfreeprob failed, error code %d.\n", status);
    }
}

/* Free up the CPLEX environment, if necessary */

if ( env != NULL ) {
    status = CPXcloseCPLEX (&env);

    /* Note that CPXcloseCPLEX produces no output,
    so the only way to see the cause of the error is to use
    CPXgeterrorstring. For other CPLEX routines, the errors will
    be seen if the CPX_PARAM_SCRIND indicator is set to CPX_ON. */

    if ( status ) {
        char errmsg[1024];
        fprintf (stderr, "Could not close CPLEX environment.\n");
        CPXgeterrorstring (env, status, errmsg);
        fprintf (stderr, "%s", errmsg);
    }
}

}
```





```

/* Free up the problem data arrays, if necessary. */

free_and_null ((char **) &probrname);
free_and_null ((char **) &obj);
free_and_null ((char **) &rhs);
free_and_null ((char **) &sense);
free_and_null ((char **) &matbeg);
free_and_null ((char **) &matcnt);
free_and_null ((char **) &matind);
free_and_null ((char **) &matval);
free_and_null ((char **) &lb);
free_and_null ((char **) &ub);
free_and_null ((char **) &ctype);

return (status);

} /* END main */

#ifdef CPX_PROTOTYPE_MIN
static int
setproblemdata (char **probrname_p, int *numcols_p, int *numrows_p,
                int *objsen_p, double **obj_p, double **rhs_p,
                char **sense_p, int **matbeg_p, int **matcnt_p,
                int **matind_p, double **matval_p,
                double **lb_p, double **ub_p, char **ctype_p)
#else
static int
setproblemdata (probrname_p, numcols_p, numrows_p, objsen_p, obj_p,
                rhs_p, sense_p, matbeg_p, matcnt_p, matind_p, matval_p,
                lb_p, ub_p, ctype_p)
char    **probrname_p;
int      *numcols_p;
int      *numrows_p;
int      *objsen_p;
double   **obj_p;
double   **rhs_p;
char     **sense_p;
int      **matbeg_p;
int      **matcnt_p;
int      **matind_p;
double   **matval_p;
double   **lb_p;
double   **ub_p;
char     **ctype_p;
#endif
{
    char    *zprobrname = NULL;    /* Problem name <= 16 characters */
    double   *zobj = NULL;
    double   *zrhs = NULL;
    char     *zsense = NULL;

```

```

int      *zmatbeg = NULL;
int      *zmatcnt = NULL;
int      *zmatind = NULL;
double   *zmatval = NULL;
double   *zlb = NULL;
double   *zub = NULL;
char      *zctype = NULL;
int      status = 0;
FILE *read;
int i;
float *temp;
int *temp1;

temp=(float*)malloc(1200*sizeof(float));
temp1=(int*)malloc(1200* sizeof(int));

zprobname = (char *) malloc (16 * sizeof(char));
zobj       = (double *) malloc (NUMCOLS * sizeof(double));
zrhs       = (double *) malloc (NUMROWS * sizeof(double));
zsense     = (char *) malloc (NUMROWS * sizeof(char));
zmatbeg    = (int *) malloc (NUMCOLS * sizeof(int));
zmatcnt    = (int *) malloc (NUMCOLS * sizeof(int));
zmatind    = (int *) malloc (NUMNZ * sizeof(int));
zmatval    = (double *) malloc (NUMNZ * sizeof(double));
zlb        = (double *) malloc (NUMCOLS * sizeof(double));
zub        = (double *) malloc (NUMCOLS * sizeof(double));
zctype     = (char *) malloc (NUMCOLS * sizeof(char));

if ( zprobname == NULL || zobj      == NULL ||
    zrhs      == NULL || zsense   == NULL ||
    zmatbeg   == NULL || zmatcnt  == NULL ||
    zmatind   == NULL || zmatval  == NULL ||
    zlb       == NULL || zub      == NULL ||
    zctype    == NULL
    ) {
    status = 1;
    goto TERMINATE;
}

strcpy (zprobname, "example");

/* The code is formatted to make a visual correspondence
   between the mathematical linear program and the specific data
   items.   */

printf("===== zobj =====\n");

read=fopen("d:\\data\\obj.txt","r");
for(i=0;i<300;i++)
{
    fscanf(read,"%f",&temp[i]);
}
for(i=0;i<300;i++)
{

```

```
        zobj[i]=temp[i];
        printf("%.1f",zobj[i]);
    }
    printf("\n");
    fclose(read);

printf("===== zmatbeg=====\\n");

read=fopen("d:\\data\\zmatbeg.txt","r");
for(i=0;i<300;i++)
{
    fscanf(read,"%d",&temp1[i]);
}
for(i=0;i<300;i++)
{
    zmatbeg[i]=temp1[i];
    printf("%d",zmatbeg[i]);
}
printf("\n");
fclose(read);

printf("===== zmatcnt =====\\n");

read=fopen("d:\\data\\zmatcnt.txt","r");
for(i=0;i<300;i++)
{
    fscanf(read,"%d",&temp1[i]);
}
for(i=0;i<300;i++)
{
    zmatcnt[i]=temp1[i];
    printf("%d",zmatcnt[i]);
}
printf("\n");
fclose(read);

printf("===== zmatind =====\\n");

read=fopen("d:\\data\\zmatind.txt","r");
for(i=0;i<1108;i++)
{
    fscanf(read,"%d",&temp1[i]);
}
for(i=0;i<1108;i++)
{
    zmatind[i]=temp1[i];
    printf("%d",zmatind[i]);
}
```

```
printf("\n");
fclose(read);

printf("===== zmatval =====\n");

read=fopen("d:\\data\\zmatval.txt","r");
for(i=0;i<1108;i++)
{
    fscanf(read,"%f",&temp[i]);
}
for(i=0;i<1108;i++)
{
    zmatval[i]=temp[i];
    printf(" %2.1f",zmatval[i]);
}
printf("\n");
fclose(read);

printf("===== zlb =====\n");

for(i=0;i<300;i++)
{
    zlb[i]=0.0;
    printf("%2.1f",zlb[i]);
}

printf("\n");

printf("===== zub =====\n");

for(i=0;i<12;i++)
{
    zub[i]=CPX_INFBOUND;
    printf("%c",zub[i]);
}

printf("\n");

for(i=12;i<18;i++)
{
    zub[i]=450000;
    printf("%2.1f",zub[i]);
}

printf("\n");

for(i=18;i<30;i++)
{
    zub[i]=CPX_INFBOUND;
    printf("%c",zub[i]);
}

printf("\n");
```



```
        for(i=30;i<36;i++)
    {
        zub[i]=260000;
        printf("%2.1f",zub[i]);
    }

    printf("\n");

    for(i=36;i<48;i++)
    {
        zub[i]=CPX_INFBOUND;
        printf("%c",zub[i]);
    }

    printf("\n");

    for(i=48;i<54;i++)
    {
        zub[i]=450000;
        printf("%2.1f",zub[i]);
    }

    printf("\n");

    for(i=54;i<66;i++)
    {
        zub[i]=CPX_INFBOUND;
        printf("%c",zub[i]);
    }

    printf("\n");

    for(i=66;i<72;i++)
    {
        zub[i]=450000;
        printf("%2.1f",zub[i]);
    }

    printf("\n");

    for(i=72;i<84;i++)
    {
        zub[i]=CPX_INFBOUND;
        printf("%c",zub[i]);
    }

    printf("\n");

    for(i=84;i<90;i++)
    {
        zub[i]=250000;
        printf("%2.1f",zub[i]);
    }

    printf("\n");

    for(i=90;i<102;i++)
    {
        zub[i]=CPX_INFBOUND;
```



```
        printf("%c",zub[i]);
    }

    printf("\n");

    for(i=102;i<108;i++)
    {
        zub[i]=450000;
        printf("%.21f",zub[i]);
    }

    printf("\n");

    for(i=108;i<114;i++)
    {
        zub[i]=450000;
        printf("%.21f",zub[i]);
    }

    printf("\n");

    for(i=114;i<120;i++)
    {
        zub[i]=450000;
        printf("%.21f",zub[i]);
    }

    printf("\n");

    for(i=120;i<204;i++)
    {
        zub[i]=CPX_INFBOUND;
        printf("%c",zub[i]);
    }

    printf("\n");

    for(i=204;i<210;i++)
    {
        zub[i]=100000;
        printf("%.21f",zub[i]);
    }

    printf("\n");

    for(i=210;i<216;i++)
    {
        zub[i]=200000;
        printf("%.21f",zub[i]);
    }

    printf("\n");

    for(i=216;i<222;i++)
    {
        zub[i]=100;
        printf("%.21f",zub[i]);
    }

    printf("\n");
```



```
        for(i=222;i<228;i++)
        {
            zub[i]=200;
            printf("%.1f",zub[i]);
        }

        printf("\n");

        for(i=228;i<234;i++)
        {
            zub[i]=300000;
            printf("%.1f",zub[i]);
        }

        printf("\n");

        for(i=234;i<240;i++)
        {
            zub[i]=600;
            printf("%.1f",zub[i]);
        }

        printf("\n");

        for(i=240;i<300;i++)
        {
            zub[i]=CPX_INFBOUND;
            printf("%c",zub[i]);
        }

        printf("\n");

printf("===== zctype =====\n");

        for(i=0;i<120;i++)
        {
            zctype[i]='I';
            printf("%c",zctype[i]);
        }


        printf("\n");

        for(i=120;i<168;i++)
        {
            zctype[i]='C';
            printf("%c",zctype[i]);
        }

        printf("\n");

        for(i=168;i<300;i++)
        {
            zctype[i]='I';
            printf("%c",zctype[i]);
        }

        printf("\n");
```



```
/* The right-hand-side values don't fit nicely on a line above.  So put  
   them here.  */
```

```
printf("===== zsense =====\n");
```

```
    for(i=0;i<24;i++)  
    {  
        zsense[i]='L';  
        printf("%c",zsense[i]);  
    }  
    printf("\n");  
    for(i=24;i<36;i++)  
    {  
        zsense[i]='E';  
        printf("%c",zsense[i]);  
    }  
    printf("\n");  
  
    for(i=36;i<42;i++)  
    {  
        zsense[i]='L';  
        printf("%c",zsense[i]);  
    }  
    printf("\n");  
    for(i=42;i<60;i++)  
    {  
        zsense[i]='E';  
        printf("%c",zsense[i]);  
    }  
    printf("\n");  
  
    for(i=60;i<66;i++)  
    {  
        zsense[i]='L';  
        printf("%c",zsense[i]);  
    }  
    printf("\n");  
    for(i=66;i<84;i++)  
    {  
        zsense[i]='E';  
        printf("%c",zsense[i]);  
    }  
    printf("\n");  
  
    for(i=84;i<90;i++)  
    {  
        zsense[i]='L';  
        printf("%c",zsense[i]);  
    }  
    printf("\n");
```





```
for(i=90;i<108;i++)
{
    zsense[i]='E';
    printf("%c",zsense[i]);
}

    printf("\n");

for(i=108;i<114;i++)
{
    zsense[i]='L';
    printf("%c",zsense[i]);
}

    printf("\n");
for(i=114;i<132;i++)
{
    zsense[i]='E';
    printf("%c",zsense[i]);
}

    printf("\n");

for(i=132;i<138;i++)
{
    zsense[i]='L';
    printf("%c",zsense[i]);
}

    printf("\n");
for(i=138;i<156;i++)
{
    zsense[i]='E';
    printf("%c",zsense[i]);
}

    printf("\n");

for(i=156;i<162;i++)
{
    zsense[i]='L';
    printf("%c",zsense[i]);
}

    printf("\n");
for(i=162;i<180;i++)
{
    zsense[i]='E';
    printf("%c",zsense[i]);
}

    printf("\n");

for(i=180;i<264;i++)
{
    zsense[i]='L';
    printf("%c",zsense[i]);
}

    printf("\n");
```



```

        for(i=264;i<294;i++)
        {
            zsense[i]='E';
            printf("%c",zsense[i]);
        }

        printf("\n");

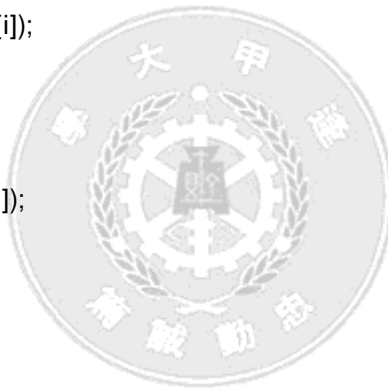
        for(i=294;i<366;i++)
        {
            zsense[i]='L';
            printf("%c",zsense[i]);
        }

        printf("\n");

printf("===== zrhs =====\n");

read=fopen("d:\\data\\zrhs.txt","r");
for(i=0;i<366;i++)
{
    fscanf(read,"%f",&temp[i]);
}
for(i=0;i<366;i++)
{
    zrhs[i]=temp[i];
    printf("%2.1f\n",zrhs[i]);
}
printf("\n");
fclose(read);

```



TERMINATE:

```

if ( status ) {
    free_and_null ((char **) &zprobname);
    free_and_null ((char **) &zobj);
    free_and_null ((char **) &zrhs);
    free_and_null ((char **) &zsense);
    free_and_null ((char **) &zmatbeg);
    free_and_null ((char **) &zmatcnt);
    free_and_null ((char **) &zmatind);
    free_and_null ((char **) &zmatval);
    free_and_null ((char **) &zlb);
    free_and_null ((char **) &zub);
    free_and_null ((char **) &zctype);
}
else {
    *numcols_p    = NUMCOLS;
    *numrows_p    = NUMROWS;
    *objsen_p     = CPX_MAX;    /* The problem is maximization */
}

```

```
*probname_p  = zprobname;
*obj_p        = zobj;
*rhs_p        = zrhs;
*sense_p      = zsense;
*matbeg_p     = zmatbeg;
*matcnt_p     = zmatcnt;
*matind_p     = zmatind;
*matval_p     = zmatval;
*lb_p         = zlb;
*ub_p         = zub;
*ctype_p      = zctype;
}
return (status);

} /* END setproblemdata */

/* This simple routine frees up the pointer *ptr, and sets *ptr to NULL */

#ifdef CPX_PROTOTYPE_MIN
static void
free_and_null (char **ptr)
#else
static void
free_and_null (ptr)
char **ptr;
#endif
{
    if ( *ptr != NULL ) {
        free ( *ptr );
        *ptr = NULL;
    }
} /* END free_and_null */
```

