

Chapter 6

模擬實驗設計與最佳化求解

- 介紹實驗設計與最佳化求解的基本觀念。
- **Process Analyzer**可以協助使用者進行實驗設計，模擬不同的參數組合以了解參數對績效的影響。
- **Optquest**根據事先設定的目標與限制條件，自動搜尋較具潛力的參數組合並進行模擬，提供最佳化的參數組合。

實驗設計

1. 了解要分析的系統對象 (水果的栽種)
 2. 建立實驗假設：可能對結果有影響的實驗控制 (品種×肥料)
 3. 有系統地調整控制參數，分別實驗並紀錄結果 (收成的質與量)
 4. 分析所有數據以判斷參數對實驗結果的影響程度 (品種<肥料)
 5. 必要時，再進行下一階段的實驗。(肥料×施肥日期)
- 一般實驗常有不可控的因素 (天氣、土壤、蟲害...)
 - 模擬實驗可以完全掌控實驗細節，沒有未知或不可控因子。

實驗設計名詞

- **因子(factors)**：分為可控因子與不可控因子，前者是實驗本身或實際上可控制改變的變數，後者是可觀察但無法控制的變數。在模擬實驗裡，分別稱為控制變數與環境變數。
- **水準(levels)**：因子可有幾種不同設定，可以是量化設定，如看板數目，也可以是質性設定，如不同的訂單排程法則。
- **配方(treatments)**：各個因子各自設定後的組合，代表一個完整的實驗設定，例如庫存管理範例的庫存上限與訂貨點。
- **反應(responses)**：以配方進行實驗後所獲得的模擬結果，通常是關鍵績效指標，如顧客等候時間。如果是質性結果，例如專案是否如期完成，則可修改量化為如期完成的機率。

不同模擬範例的因子與反應

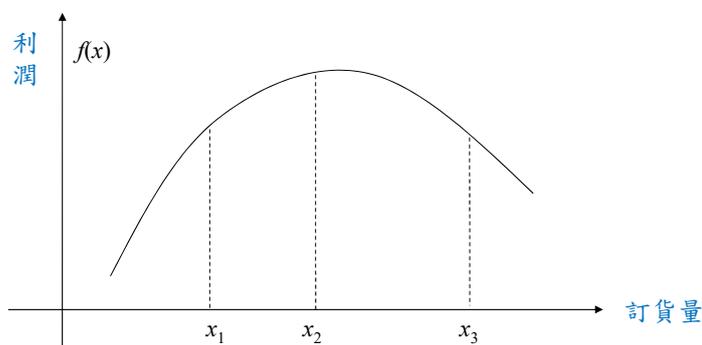
System	Factors	量化	質性	環境	Response
超級市場	顧客多寡(到達速率)	✓		✓	平均排隊人數 平均等候時間 收銀員使用率
	櫃台數目	✓			
	使用快速結帳櫃台		✓		
生產線	訂單種類組合	✓		✓	產出率 設備使用率 平均流程時間
	機器數目	✓			
	訂單排序規則		✓		
庫存控管	需求多寡(到達速率)	✓		✓	平均持有成本 平均缺貨成本 平均訂購成本 供貨水準
	再訂購點	✓			
	補貨數量	✓			
	補貨前置時間	✓		✓	
	銷售資訊分享		✓		

應用實驗設計於模擬實驗

- (0) 決定實驗因子數目與水準設定 (不要貪心)
 - (1) 根據因子數目與水準數目設定實驗設計的各個配方
 - (2) 對各個配方進行模擬，紀錄反應，也就是模擬產生的績效
 - (3) 比較各個配方，估算各個因子對於反應的影響幅度
- 模擬特有的優點：所有因子都能控制，能完美複製實驗結果
 - 模擬同樣會有的缺點：隨機誤差

單因子實驗設計

分析單一因子的調整是否對績效有顯著影響



應模擬三個以上的水準，以觀察兩者之間的關係變化

雙因子實驗設計

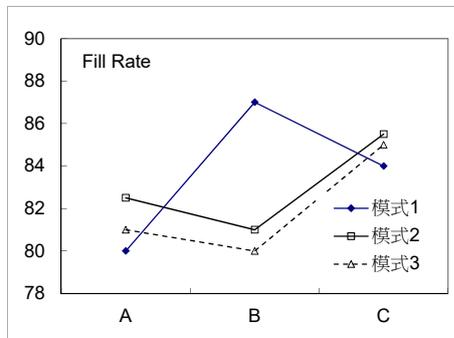
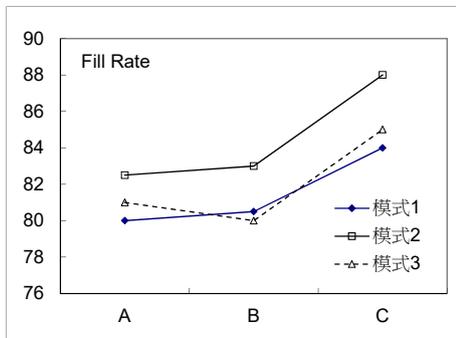
	因子A	因子B	Interaction
配方1	-1	-1	+1
配方2	-1	+1	-1
配方3	+1	-1	-1
配方4	+1	+1	+1

因子A的主效用 $\frac{(Y_3 - Y_1) + (Y_4 - Y_2)}{2}$

AB交互作用 $\frac{(Y_4 - Y_3) - (Y_2 - Y_1)}{2} = \frac{(Y_1 + Y_4)}{2} - \frac{(Y_2 + Y_3)}{2}$

主效用 vs. 交互作用

庫存控管模式(1, 2, 3) × 安全庫存水準(A, B, C)



部分因子設計與直交設計

$2^3 \rightarrow 2^2$

	因子A	因子B	因子C
配方1	-1	-1	-1
配方2	+1	+1	-1
配方3	+1	-1	+1
配方4	-1	+1	+1

$2^7 \rightarrow 2^3$

A	B	C	D	E	F	G
-1	-1	-1	-1	-1	-1	-1
-1	-1	+1	-1	+1	+1	+1
-1	+1	-1	+1	+1	-1	+1
-1	+1	+1	+1	-1	+1	-1
+1	-1	-1	+1	-1	+1	+1
+1	-1	+1	+1	+1	-1	-1
+1	+1	-1	-1	+1	+1	-1
+1	+1	+1	-1	-1	-1	+1

Orthogonal Design: 任何一行的總和為零，
且任何兩行兩兩相乘的和(向量內積)均為0

看板系統的完整因子設計

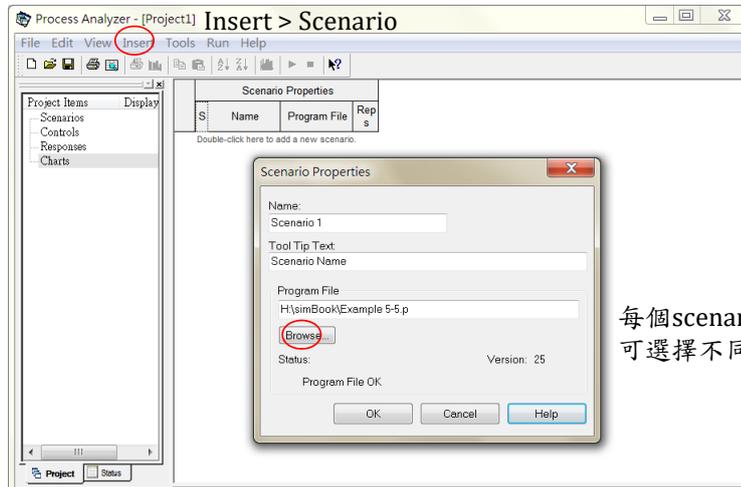
Example 5-5.doe: 評估哪個控制參數對於產出率的影響最大

num_A	num_B	Kanban 2
2	2	2
2	2	3
2	3	2
2	3	3
3	2	2
3	2	3
3	3	2
3	3	3

Response: 系統的產出率 output rate

PAN: Process Analyzer

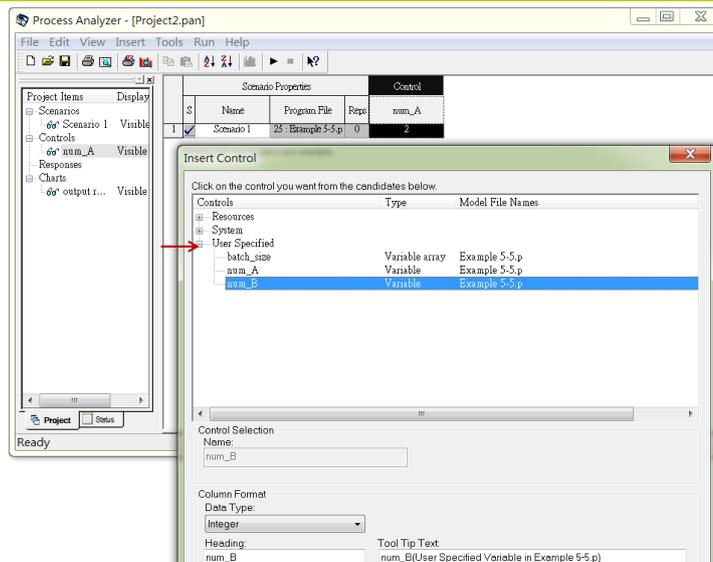
Arena: Tools > Process Analyzer



每個scenario(treatment)
可選擇不同的程式

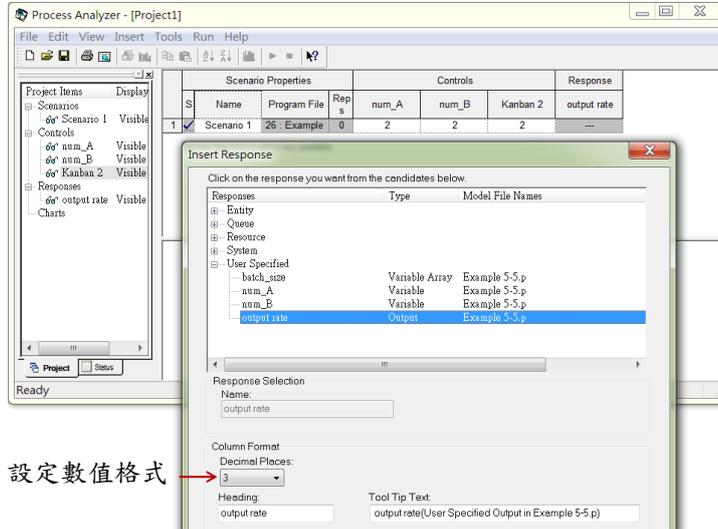
選擇可控制因子

Insert > Control

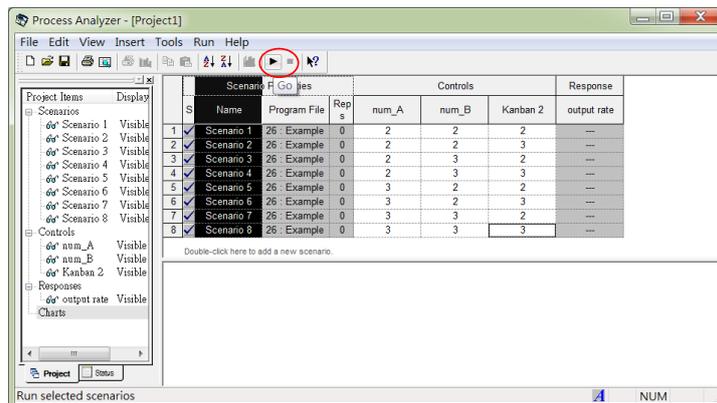


選擇要比較的績效

Insert > Response



設定完整的 2^3 因子設計



1. 滑鼠選擇所有 Scenarios，按下 Go 按鈕並確認
2. 副檔名為 *.pan

PAN自動執行模擬

	num_A	num_B	Kanban 2	Output Rate	AB Interaction
1	-1	-1	-1	0.584	+1
2	-1	-1	+1	0.598	+1
3	-1	+1	-1	0.585	-1
4	-1	+1	+1	0.595	-1
5	+1	-1	-1	0.621	-1
6	+1	-1	+1	0.626	-1
7	+1	+1	-1	0.610	+1
8	+1	+1	+1	0.615	+1

$$\frac{(0.621 + 0.626 + 0.610 + 0.615)}{4} - \frac{(0.584 + 0.598 + 0.585 + 0.595)}{4} = 0.0275$$

$$\left(\frac{0.584 + 0.598 + 0.610 + 0.615}{4} - \frac{0.585 + 0.595 + 0.621 + 0.626}{4} \right) = -0.005$$

Process Analyzer主要步驟

- 在Arena內部啟動Process Analyzer。
- 建立新的Project，或是叫出舊有的Project進行修改。
- 從Insert選單選擇Insert Scenario，接著找出程式的執行檔。
- 從Insert選單選擇Insert Control，也就是實驗設計的因子，一般都是資源數量或是使用者定義的變數。
- 從Insert選單選擇Insert Response，在跳出的視窗內選擇你要分析的績效。
- 陸續加入新的Scenario，根據實驗設計去調整各因子的水準。
- 選擇所有Scenarios，然後按下▶按鈕，讓Process Analyzer進行模擬。

PAN的多重比較功能

用滑鼠選擇Response欄，從Insert選單裡選擇加入Chart

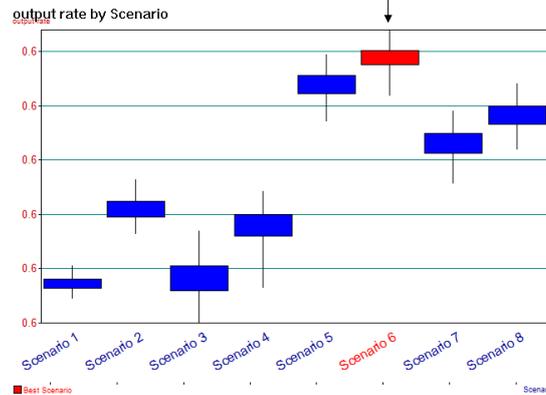
選擇Box and Whisker盒鬚圖

設定績效值越高越好

Identify Best Scenarios
Bigger is better

多重比較判定明顯較佳的Scenario

Scenario 6: num_A=3, num_B=2, Kanban 2=3



6.1

Summary 模擬與實驗設計

- 模擬實驗幾乎可完全掌控實驗的細節，可以完美複製結果，不受到外界環境的干擾。
- 因子水準設定不當或是模擬誤差太大，可能造成無效的實驗，無法正確分析因子的影響。
- 改善可能來自因子與其他因子的交互作用，或是其他未注意的因素，而不是實驗因子本身的影響。
- 初步實驗設計可不考慮較不重要的因子，以節省時間與成本。另一選擇是採用部分因子設計或是直交設計的實驗。

產品組合最佳化問題

- 決策是兩種產品的產量， x_1 與 x_2 ，追求利潤的最大化

兩種產品每單位產量各耗費1單位的原料A
各需要2單位與3單位的原料B

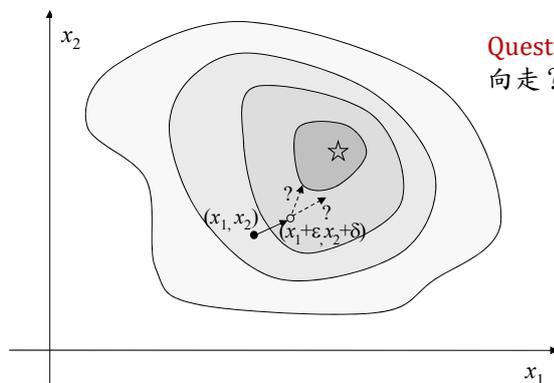
Maximize $f(x_1, x_2)$
subject to $x_1 + x_2 \leq 6$
 $2x_1 + 3x_2 \leq 10$
 $x_1, x_2 \geq 0$

原料A的供應只有6單位
原料B的供應只有10單位

- 模擬最佳化：無法直接計算目標函數 f ，必須將 x_1 與 x_2 的設定值輸入模擬程式，模擬後才能得到 $f(x_1, x_2)$ 的估計值。
- 有智慧地調整 x_1 與 x_2 的值，再進行模擬，期望能改善目標值

梯度法往高的方向尋找最佳解的位置

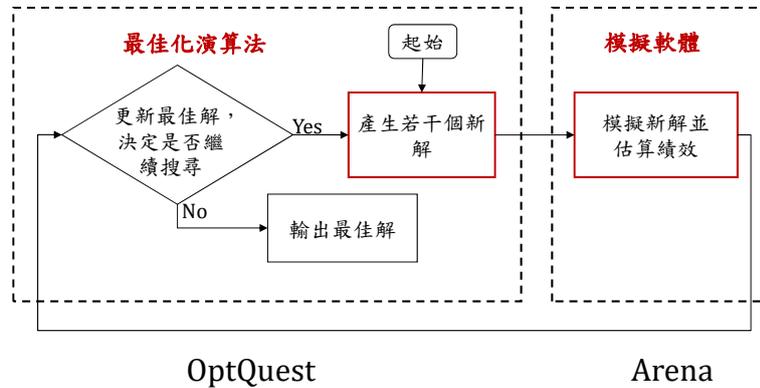
模擬結果顯示 $f(x_1+\varepsilon, x_2+\delta)$ 優於 $f(x_1, x_2)$



Questions: 下一步該往哪個方向走？繼續同方向？走多遠？

Question: 如果 $f(x_1+\varepsilon, x_2+\delta)$ 比較差呢？

最佳化演算法與模擬的結合應用

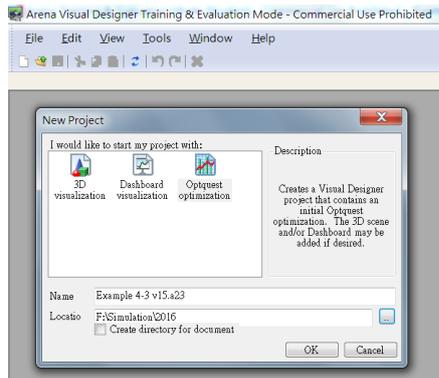


模擬最佳化的挑戰

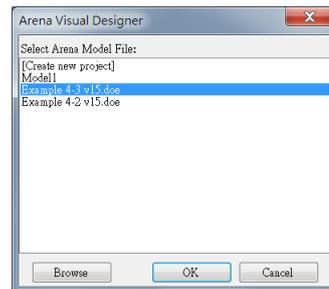
- 模擬最佳化需要一個好的演算法去搜尋優良解，否則無異於大海撈針
- 模擬最佳化的運算時間包括尋找新的解以及進行模擬評估所需的時間。
- 如果每次模擬評估的時間太短，模擬誤差就會增大，搜尋無異於散彈打鳥。如果加長模擬時間以提高模擬準確度，則在有限時間內，演算法搜尋的速度慢，不易找出好的解
- 模擬最佳化所得到的可能是區域最佳解，而不是全域最佳解，也無法得知現有最佳解與真實最佳解的差距。

OptQuest for Arena

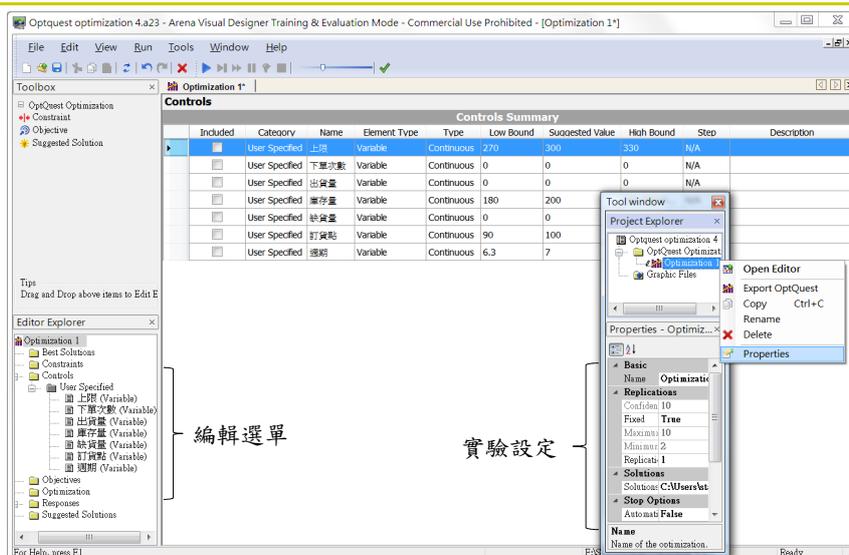
1. 開啟 Example 4-3.doe , Tools > OptQuest for Arena
2. File > New Project
3. 在 New Project 視窗內選擇 Optquest optimization



4. 選擇對應的 Arena 程式



OptQuest 操作介面



How OptQuest Controls Arena

- 先打開要最佳化的Arena程式，再啟動OptQuest
- OptQuest會完全控制Arena，列出程式內所有可控制的參數(**controls**)與已定義的績效(**responses**)
- 經由編輯選單(editor explorer)，設定各個參數的實驗範圍與相關的績效
- 經由編輯選單，使用者設定限制式(**constraints**)與最佳化的目標績效(**objective**)
- Project Explorer設定實驗的搜尋次數或停止搜尋的條件
- When the optimization runs:
 - OptQuest 控制Arena程式使用參數的建議值，開始進行模擬
 - Arena 將模擬結果交給OptQuest評估
 - OptQuest根據搜尋範圍與累積的模擬結果，以智慧型演算法產生新的一組參數，輸入Arena進行模擬。
 - 整個過程不斷重複，直到目標績效符合實驗條件
- OptQuest 結束後，Arena程式回復原先設定

設定參數的搜尋範圍

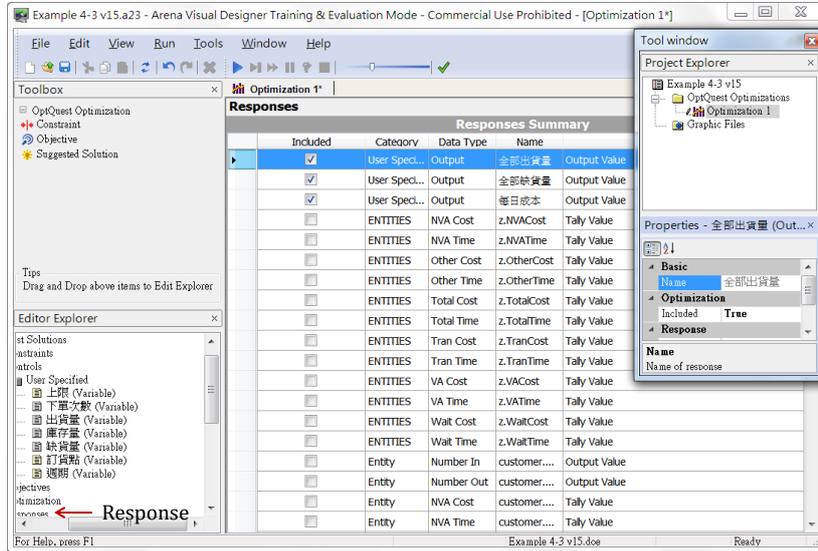
設定控制參數
的實驗範圍

Controls						Controls Summary			
Included	Category	Name	Element Type	Type	Low Bound	Suggested	High Bound	Step	
<input checked="" type="checkbox"/>	User Specified	上限	Variable	Discrete	200	300	800	10	
<input type="checkbox"/>	User Specified	下單次數	Variable	Continuous	0	0	0	N/A	
<input type="checkbox"/>	User Specified	出貨量	Variable	Continuous	0	0	0	N/A	
<input type="checkbox"/>	User Specified	庫存量	Variable	Continuous	180	200	220.000000...	N/A	
<input type="checkbox"/>	User Specified	缺貨量	Variable	Continuous	0	0	0	N/A	
<input checked="" type="checkbox"/>	User Specified	訂貨點	Variable	Discrete	100	100	400	10	
<input checked="" type="checkbox"/>	User Specified	運期	Variable	Discrete	1	7	40	1	

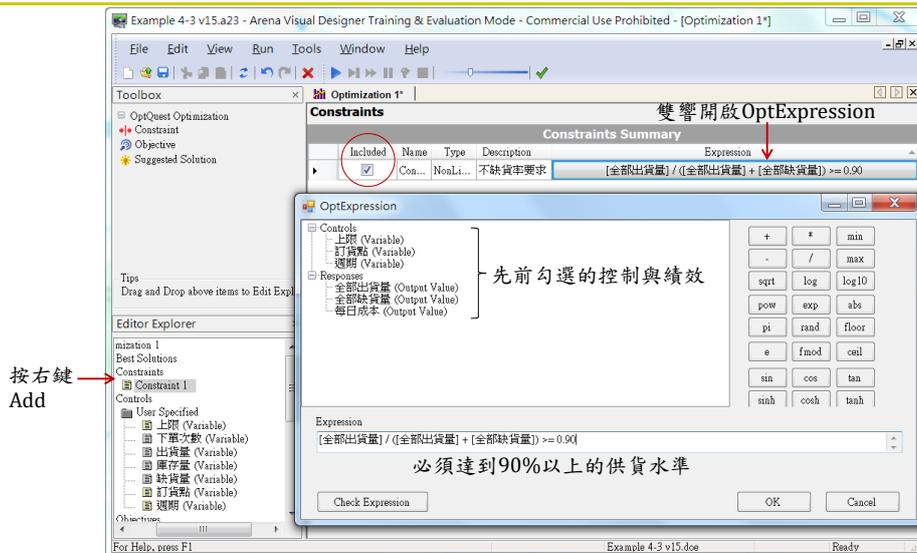
$$\frac{\text{High Bound} - \text{Low Bound}}{\text{Step}}$$

Discrete選項可以設定Step Size

勾選Optquest要追蹤的績效



設定限制式



設定最佳化目標

按右鍵 Add

雙響開啟OptExpression

Name	Type	Goal	Description	Expression
Objective 1	Non-Linear	Minimize	成本最小化	[每日成本]

OptExpression Dialog Contents:

- Controls: 上升 (Variable), 打掃點 (Variable), 送解 (Variable)
- Responses: 全部出貨量 (Output Value), 全部裝貨量 (Output Value), 每日成本 (Output Value)

Properties - Objectives Summary:

- Name: Objective 1
- Description: 成本最小化
- Expression: [每日成本]
- Goal: Minimize
- Type: NONLINEAR

NKFUST Simulation Chapter 6 – Experimental Design and Optimization Slide 31

實驗條件設定

1 動態調整每個解的模擬次數

2 最佳化過程存為*.log檔

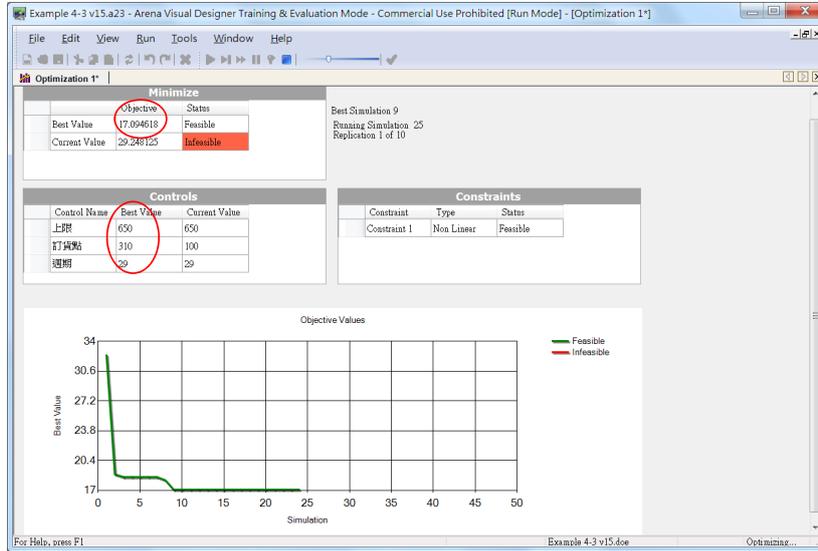
3 搜尋300個解後結束

Property	Value
Name	Optimization 1
Confidence Interval Percent	10
Fixed	False
Maximum Replications	100
Minimum Replications	10
Replications	1
Solutions Log	C:\Users\staff\Documents\OptQuest
Automatic Stop	False
Manual Stop	False
Number of Simulations	300
Run Only Suggested Solutions	False
Use Number Of Simulations	True
Tolerance	True
Tolerance	False

Use Number Of Simulations: Indicates the optimization will stop after Number of Simulations is reached.

NKFUST Simulation Chapter 6 – Experimental Design and Optimization Slide 32

最佳化過程的初期進展

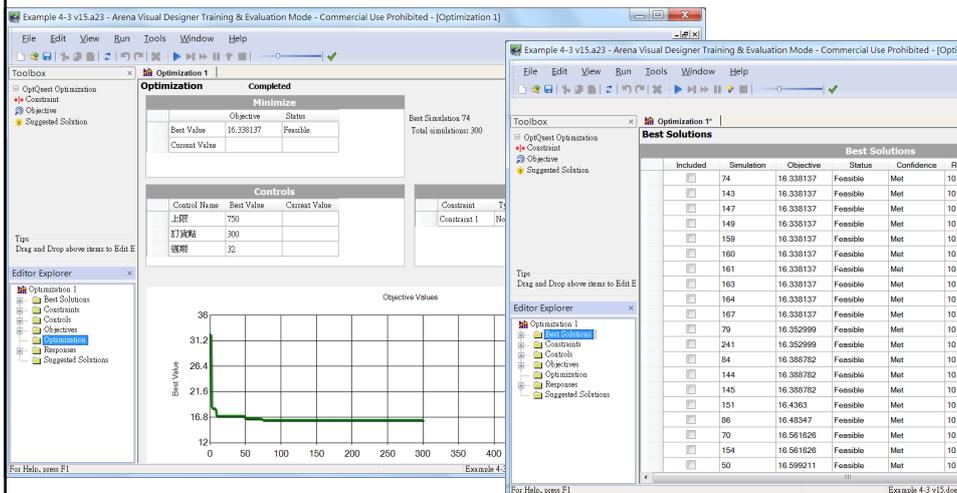


NKFUST Simulation

Chapter 6 – Experimental Design and Optimization

Slide 33

Optquest列出最佳解



Optquest副檔名為*.a23，舊版副檔名為*.opt

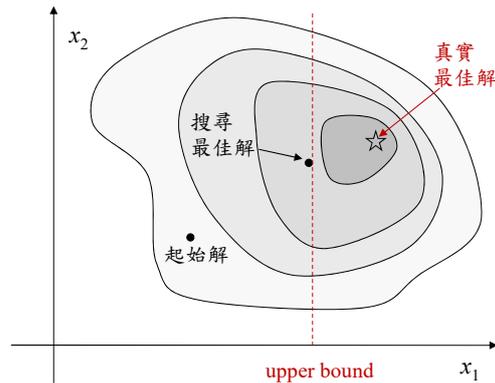
NKFUST Simulation

Chapter 6 – Experimental Design and Optimization

Slide 34

搜尋最佳解失敗的原因

- 搜尋範圍太小，或是Step Size太大，可能錯失優良解。
- 變數過多、搜尋範圍過大、或是Step Size過小，將使搜尋過程如同大海撈針，在有限時間內可能找不到品質優良的解。



- 搜尋範圍受限，無法跨越界限尋找更好的解。

Summary 模擬最佳化

- 模擬最佳化要花費時間尋找新的解以及進行模擬評估解的好壞，最佳化過程非常費時。
- 模擬次數少，誤差可能會影響最佳解的判斷，模擬次數多，結果會較精確，但是最佳化過程可能進行緩慢。
- 小心選擇變數與設定搜尋範圍，以免事倍功半。
- Optquest在設定條件下搜尋到的最佳解不保證是真正最佳解
- 可以現有最佳解為suggested value進行第二次搜尋，移動或縮小搜尋範圍、或縮小step size，以確認現有最佳解的品質，或是逼近真正的全域最佳解。