



 國立高雄第一科技大學

運用多目標模擬最佳化演算法 求解生產控制與庫存管理問題

A Multi-Objective Evolutionary Algorithm for Simulation Optimization of Production and Inventory Control Problems

指導教授：郭幸民 博士
 研究生：吳心怡

● 前言 ● 文獻探討 ● 研究設計 ● 數學測試 ● 模擬實驗 ● 結論與建議

研究背景

- 真實世界的管理問題難以用數學方法求得理論解。
→ 模擬成為分析複雜系統的最佳工具。
- 傳統上模擬被視為績效評估的工具，難以追求系統設定之最佳化。
- 結合模擬與最佳化方法將可提供完整的研究工具。

● 前言 ● 文獻探討 ● 研究設計 ● 數學測試 ● 模擬實驗 ● 結論與建議

研究動機

- 目前模擬最佳化方法大多應用在單目標問題，但是真實問題通常牽涉到多個相互衝突的目標。
- 處理多目標最佳化問題的方式：
 - 利用權重值或其他公式將多個績效指標結合成單一目標問題來進行最佳化。
 - 在同時考慮多個目標的前提下，應用最佳化演算法搜尋所有的非凌駕解(non-dominated solutions)。
- 本研究發展適合模擬實驗的多目標最佳化演算法，在模擬實驗費時的過程中，提升實驗效率。

● 前言 ● 文獻探討 ● 研究設計 ● 數學測試 ● 模擬實驗 ● 結論與建議

研究目的

- 本研究提出適合模擬實驗的多目標最佳化演算法，求解生產控制與庫存管理問題。
 - 參考每次的模擬實驗結果，利用不同的Selection方法對每一個目標產生一個mating pool。
 - 以不同方式將不同mating pool成員進行交配。
 - 在每一代模擬實驗結束後，對非凌駕解集合執行更新的動作。
 - 測試模擬誤差幅度對本研究提出的演算法求解品質與效率之影響程度。

● 前言 ● 文獻探討 ● 研究設計 ● 數學測試 ● 模擬實驗 ● 結論與建議

模擬最佳化方法的發展

- Tekin and Sabuncuoglu(2004) 提到已經有許多的技術與方法來解決模擬最佳化問題：
 - 傳統的方法：Ranking and Selection、Random Search與Response Surface Methodology等，只能求區域最佳解。
 - 近年來的新方法：Evolutionary Algorithms、Tabu Search與Simulated Annealing等方法，可求得全域最佳解。
 - 結合最佳化功能於模擬程式中可改善過去只能將模擬作為系統評估工具的限制。

● 前言 ● 文獻探討 ● 研究設計 ● 數學測試 ● 模擬實驗 ● 結論與建議

最佳化軟體

Package	Simulation software products supported	Search procedures used
AutoState	AutoMod, AutoSched	Evolution strategies
OptQuest	ARENA, Micro Saint, QUEST	Scatter search, Tabu search, Neural networks
OPTIMIZ	SIMUL8	Neural networks
SimRunner2	MedModel, ProModel	Evolution strategies, Genetic algorithm
WITNESS Optimizer	WITNESS	Simulated annealing, Tabu search

Source: Law and Kelton (2000)

模擬最佳化的演算法

- Pichitlamken and Nelson(2002) 提出的演算法具有下列特色：
 - 運用 Global guidance system 確保實驗不陷入區域最佳解。在時間允許的前提下，實驗可收斂並最少找到一個最佳解。
 - 應用 Selection-of-the-best 步驟來減少模擬實驗的次數，卻不影響解的品質。
 - 使用 Local improvement 改善實驗結果的穩定性，讓搜尋能集中於較佳解的分布區域。

7

模擬最佳化的演算法

- Boesel *et al.*(2003) 將 ranking-and-selection 方法引進基因演算法，在系統運作結束後，根據模擬平均值將較佳解分類，必要時增加額外的模擬實驗次數，以確保選擇較佳解過程的準確性。
- Rosen and Harmonosky(2005) 運用模擬退火法求解離散型的模擬最佳化問題，利用實驗過程產生的情報來輔助搜尋，以改善演算法的收斂速度。

8

多目標最佳化

$$\begin{aligned} \text{Max or Min } Z(x_1, \dots, x_n) &= \{f_1(x_1, \dots, x_n), f_2(x_1, \dots, x_n), \dots, f_m(x_1, \dots, x_n)\} \\ \text{subject to } g_i(x_1, \dots, x_n) &\leq b_i \quad i=1, \dots, n \quad (\text{Resource Constraints}) \\ h_j(x_1, \dots, x_n) &\geq c_j \quad j=1, \dots, m \quad (\text{Performance Requirements}) \\ x_i &\geq 0 \end{aligned}$$

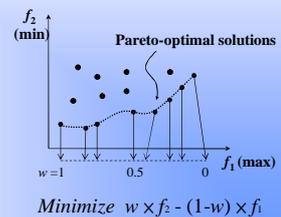
- 加權法 (Weighted Sum Method)：事先對每一目標給定權重，將各個目標加權合併為單一目標。
- ϵ -限制法 (ϵ -Constraint Method)：追求某單一目標的最佳化，同時限制其他目標在特定誤差內。
- 目標規劃法 (Goal Programming Method)：對每一個目標建立一目標值，尋求與每個目標值之差距 (Deviation) 總和最小的解。

9

Pareto-optimal Solution

- 多目標問題無單一最佳解，僅能考慮非凌駕解。
 $f_i(\mathbf{x}) \leq_{\text{better}} f_i(\mathbf{y})$ for every f_i and $f_j(\mathbf{x}) <_{\text{better}} f_j(\mathbf{y})$ for some f_j
 $\Rightarrow \mathbf{x}$ is dominated by \mathbf{y}

- 利用權重組合所有考慮的目標，隨意變動權重組合找出對應的 Pareto-optimal Solution。



10

多目標模擬最佳化

- Baesler and Sepúlveda (2000) 利用具隨機性的模擬模式來結合基因演算法與目標規劃法去解決多目標模擬最佳化問題。
- Alberto *et al.*(2002) 利用加權目標規劃法將多目標問題轉換成單一目標問題，並發展演化式演算法去解決廢棄車輛場設計與管理的最佳化問題。
- Gupta and Sivakumar(2002) 研究半導體產業的多目標生產排程問題，利用加權法、全面評準近似法、最小離差法與妥協規劃法去搜尋非凌駕解。
- Joines *et al.*(2002) 發展多目標基因演算法並結合權重法、目標規劃法與 ϵ -限制法去估計供應鏈最佳化搜源策略的 Pareto-optimal Frontier。

11

Evolutionary Algorithm

- EA 的種類包括 GA、ES、EP、GP，這些方法都擁有的共同特性：
 - 每一代都由多個解所組成 (適合用來解決多目標最佳化的問題)。
 - 利用 Selection 保留較好的母代在 mating pool。
 - 將留下的母代利用 Crossover 或 Mutation 去產生新的子代。
 - 利用 Fitness Value 決定繼續生存的族群。
 - 反覆實驗一直到達到 Stopping Criterion 才停止。

12

演化式演算法的應用

- 許多求解單目標問題的演算法被修改以求解多目標最佳化問題。
- Deb(2001) 提到第一個用來求解多目標問題的演算法為Schaffer(1984)提出的vector evaluated GA (VEGA)。
- Joines *et al.*(2002) 認為由 Deb and Goel (2002)提出的 Nondominating Sorting GA (NSGA-II) 是搜尋 Pareto-optimal solutions 最好的方法。

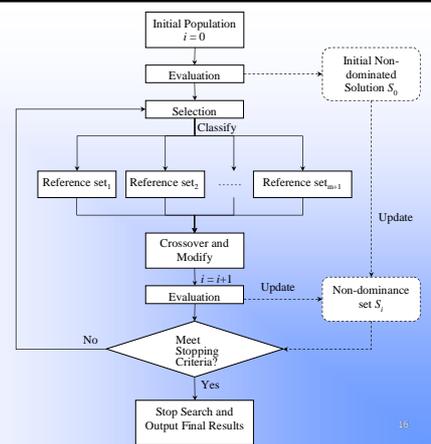
小結

1. 模擬費時與隨機特性使模擬最佳化方法大多應用於單目標問題。
2. 多目標問題通常應用確定性(deterministic)的最佳化演算法，未考慮模擬實驗的誤差。
3. EA單一次實驗可找出多個非凌駕解的特性，適合解決多目標問題。
4. 本研究發展一個簡單好用又兼顧求解效率的演算法來解決不同類型的多目標模擬問題。

演算法架構

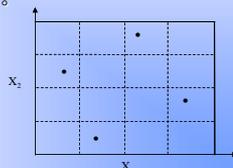
- Step 0. 利用 Latin-Hypercube Sampling (LHS) 建構起始解集合，對集合中所有的系統進行模擬實驗並找出初始的非凌駕解集合 S_0 ，令 $i=1$ 。
- Step 1. 根據每個目標函數的績效表現將母體分類為 $m+1$ 個 Reference Sets。
- Step 2. 將 Reference Sets 中的成員進行 Crossover operation 產生新的系統設計。必要時，根據資源的限制來修改系統設計中每個參數的上下界。
- Step 3. 模擬子代中所有的系統設計後，進行 Elimination operation 決定新的非凌駕解集合並且跟先前的非凌駕解集合執行更新的動作，產生非凌駕解集合 S_i 。
- Step 4. 如果目前的結果達到特定停止準則，則停止搜尋。否則，讓 $i \leftarrow i+1$ 並且回到 Step 1 重新開始。

演算法流程圖



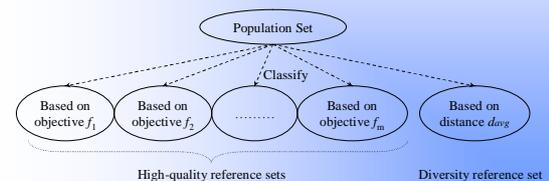
起始解建構方式

- 任意選擇：應用在初步測試階段，用來了解演算流程的可行性。
- Latin Hypercube Sampling：將每個決策變數的範圍分成 n 個等分的子範圍，執行 n 次選擇，每一個子範圍只會被選中一次。



選擇 Reference Sets 的方式

- 本研究結合 VEGA 與 Scatter Search 的概念發展出選擇 Reference Sets 的方法
- High-quality Reference Sets 與 Diversity Reference Set



High-quality Reference Sets

- 選擇方式： **d 值愈小**表示該系統設計績效表現愈接近目前的最佳解。

$$d_i(x) = w_1 \left(\frac{f_1(x) - f_1^*}{f_1^*} \right)^2 + w_2 \left(\frac{f_2(x) - f_2^*}{f_2^*} \right)^2 + \dots + w_m \left(\frac{f_m(x) - f_m^*}{f_m^*} \right)^2$$

- 假設 w_i 與 f_i^* 為第 i 個目標的權重與目前的最佳解，利用模擬實驗後每個目標的平均值 f_i 。
- 採用不同的權重設定方式
 - 令 $w_i = 1$ 與 $w_j = 0$ ，當 $j \neq i$ 。
 - 令 $w_i = 3/4$ 與 $w_j = 1/4$ ，當 $j \neq i$ 。
 - 令 $w_i = 2/3$ 與 $w_j = 1/3$ ，當 $j \neq i$ 。

19

Diversity Reference Set

- 修改 Zitzler *et al.* (2000) 提出測量演算法收斂程度的指標，做為選擇 Diversity Reference Set 的依據。
- 計算 High-quality Reference Sets 選完後，剩餘的系統設計與目前所有非凌駕解的平均距離 d_{avg} 。

$$d_{avg}(y_s) = \frac{\sum_{y' \in Y^*} \|y_s - y'\|}{|Y^*|}$$

- 選擇方式： **d_{avg} 值愈大**表示離目前非凌駕解集合愈遠，用來增加下一代交配組合的多元性。

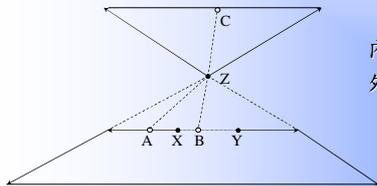
20

Crossover Operation

- 交配方案：

兩點交配： $w_a \cdot (x_1, x_2, \dots, x_m) + (1 - w_a) \cdot (y_1, y_2, \dots, y_m)$

三點交配： $w_b [w_a \bar{x} + (1 - w_a) \bar{y}] + (1 - w_b) \bar{z}$



內插法： $0 < w < 1$
外插法： $-1 < w < 2$

Scatter Search by
Glover *et al.* (2000)

21

交配類型與總組合數

交配類型	總組合數
Type1. 每個 High-quality Reference Set 的第一名進行兩點交配	$\binom{m}{2}$
Type2. 每個 High-quality Reference Set 的第一名與 Diversity Reference set 進行兩點交配	$m \times m_d$
Type3. 每個 High-quality Reference Set 的第一名與每個 High-quality Reference Set 的其他名進行兩點交配	$\binom{m}{1} \times \binom{m-1}{1} \times \binom{m-1}{1}$
Type4. 每個 High-quality Reference Set 的其他名之間互相進行兩點交配	$\binom{m}{2} \times (m-1) \times (m-1)$
Type5. 任選兩個 High-quality Reference Sets 的成員與 Diversity Reference Set 進行三點交配	$\binom{m}{2} \times m_r \times m_d$

22

Elimination Operation

- 假設目標為 Max f_1 與 Min f_2 ：

	目標 f_1	目標 f_2
系統 A	$[a_{11}, a_{12}]$	$[a_{21}, a_{22}]$
系統 B	$[b_{11}, b_{12}]$	$[b_{21}, b_{22}]$

只要其中一個假設成立就可宣稱系統 A 凌駕系統 B：

(1) $a_{11} > b_{12}$ and $a_{21} < b_{22}$

(2) $a_{12} > b_{11}$ and $a_{22} < b_{21}$

- 利用 Common Random Number 降低比較結果的變異性，實驗結果透過 Paired t-test 檢驗差距是否顯著。
- 如果仍無法分辨優劣；利用 NSGA-II 中提出的 Crowding Distance 指標，計算無顯著差異的系統跟相鄰非凌駕解的距離，留下與相鄰非凌駕解的距離愈遠的系統設計。

23

檢視演算法優劣的指標

- 收斂指標：應用 Zitzler *et al.* (2000) 提出的 M_1^* 指標來測量演算法求出的非凌駕解集合靠近真實的 Pareto-optimal 集合的程度。

$$M_1^* = \frac{1}{|Y^*|} \sum_{y' \in Y^*} \min\{\|y' - \bar{y}\|\}$$

- 多元化指標：利用 NSGA-II 中定義的 Δ 來衡量非凌駕解分佈 diversity 的程度。

$$\Delta = \frac{d_f + d_l + \sum_{i=1}^{N-1} |d_i - \bar{d}|}{d_f + d_l + (N-1)\bar{d}}$$

24

停止準則

1. 實驗達到設定的次數。
2. 非凌駕解集合連續三代都沒有變化。

數學測試

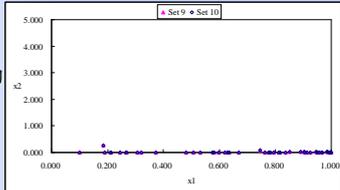
- 從多目標最佳化領域中，選擇兩種常被用來測試演算法效率的數學問題類型。
- 測試的項目包括：
 - 選擇起始解的建構方式
 - 選擇交配的方式
 - Diversity Reference Set的作用
 - 交配類型的選擇比例
 - 決定選擇Reference Sets成員的方式
 - 演算法的收斂速度與解的分佈狀況

選擇起始解的建構方式

任意選擇

$$\begin{aligned} &\text{Maximize } f_1(\mathbf{X}) = 1.1 - x_1 \\ &\text{Maximize } f_2(\mathbf{X}) = 60 - \frac{1+x_2}{x_1} \\ &\text{subject to } 0.1 \leq x_1 \leq 1 \\ &\quad \quad \quad 0 \leq x_2 \leq 5 \end{aligned}$$

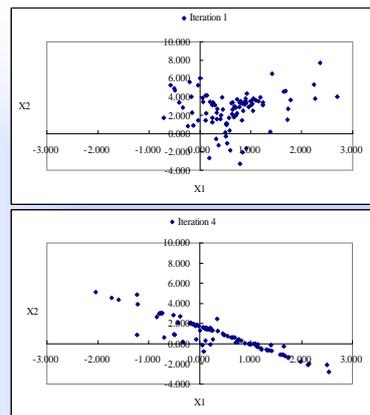
→ Pareto Optimal Solutions :
 $0.1 \leq x_1 \leq 1$ and $x_2 = 0$



結果發現：
非凌駕解集合分佈不均勻
→ 起始解的建構方式
決定採用LHS

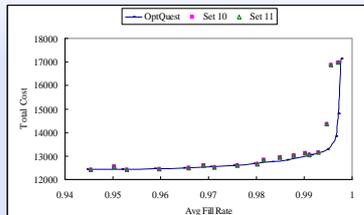
選擇交配的方式

- 兩點交配：隨著演算次數增加，搜尋的範圍逐漸變窄，陷入區域最佳解。
- 交配方式決定加入三點交配



無Diversity Reference Set的情形

初步模擬測試未建構 Diversity Reference Set，
實驗結果發現：
1. 找不到某些區段的值
2. 找不到端點的值



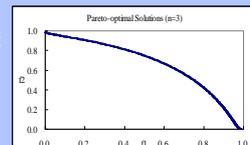
測試範例一：Fon

- Fonseca與Fleming (1995)提出無限制式的數學範例

$$\text{Fon: } \begin{cases} \text{Minimize } f_1(x) = 1 - \exp\left(-\sum_{i=1}^3 \left(x_i - \frac{1}{\sqrt{n}}\right)^2\right) \\ \text{Minimize } f_2(x) = 1 - \exp\left(-\sum_{i=1}^3 \left(x_i + \frac{1}{\sqrt{n}}\right)^2\right) \\ \text{subject to } -4 \leq x_i \leq 4 \\ \quad \quad \quad i = 1, 2, 3 \end{cases}$$

- 已知Pareto-optimal solutions :

$$x_1 = x_2 = x_3 \in [-1/\sqrt{3}, 1/\sqrt{3}]$$



Fon的實驗設定

- 利用Latin Hypercube Sampling選出十組的起始集合。
- 每次演算流程後產生兩個High-quality Reference Sets與一個Diversity Reference set，每個Reference Set成員各有三個。

	NSGA-II	本研究
Population Size	100	25
Crossover Rate	0.8	1
Mutation Rate	0.01	0
Number of Generations	250	10

31

Fon的交配類型與個數

交配類型	總組合數
Type1. 每個High-quality Reference Set的第一名進行兩點交配	1
Type2. 每個High-quality Reference Set的第一名與Diversity Reference set進行兩點交配	6
Type3. 每個High-quality Reference Set的第一名與每個High-quality Reference Set的其他名進行兩點交配	4
Type4. 每個High-quality Reference Set的其他名之間互相進行兩點交配	4
Type5. 任選兩個High-quality Reference Sets的成員與Diversity Reference Set進行三點交配	10

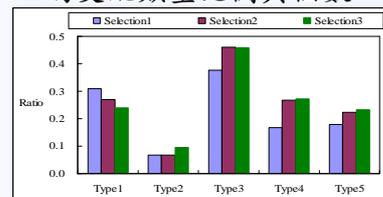
32

選擇Reference Sets成員方式

- Selection1：令 $w_i=1$ 與 $w_j=0$ ，當 $j \neq i$ 。
- Selection2：令 $w_i=3/4$ 與 $w_j=1/4$ ，當 $j \neq i$ 。
- Selection3：令 $w_i=2/3$ 與 $w_j=1/3$ ，當 $j \neq i$ 。

33

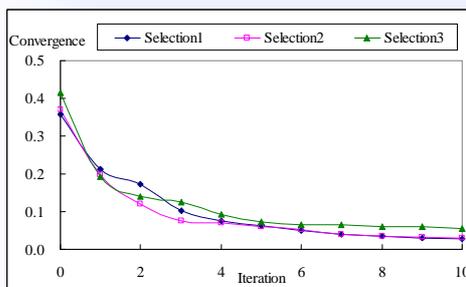
Fon：選入非凌駕解集合的交配類型比例與個數



	Type1	Type2	Type3	Type4	Type5
Selection1	3.1	4	15.1	6.7	17.8
Selection2	2.7	4.1	18.4	10.7	22.4
Selection3	2.4	5.7	18.3	10.9	23.2

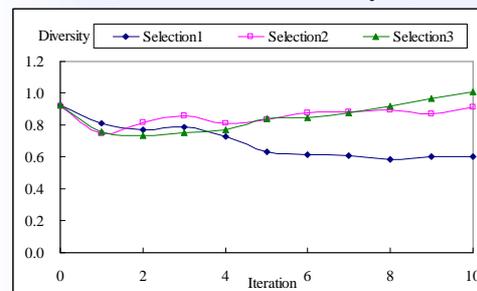
34

Fon十次實驗非凌駕解集合的收斂情形



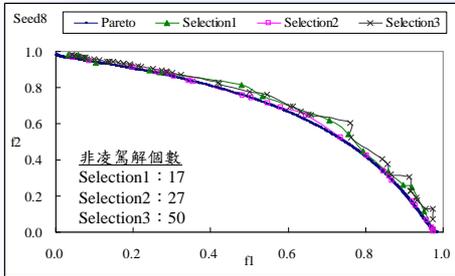
35

Fon十次實驗非凌駕解集合的Diversity情形



36

Fon第八次實驗 的非凌駕解集合分佈情形



Fon數學式收斂與多元化指標 之平均值與標準差

Algorithm	M_1^+ (Convergence)		Δ (Diversity)	
	\bar{x}	σ_n	\bar{x}	σ_n
NSGA-II	0.0019	0.0000	0.3781	0.0006
Selection1	0.0280	0.0100	0.6028	0.1008
Selection2	0.0301	0.0200	0.9138	0.2022
Selection3	0.0550	0.0444	1.0064	0.1538

測試範例二：Constr-Ex

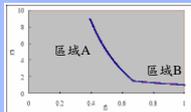
- Deb (2001) 提出有限制式的數學範例：

$$\begin{aligned} & \text{Minimize } f_1(x) = x_1 \\ & \text{Minimize } f_2(x) = \frac{1+x_2}{x_1} \\ \text{Constr-Ex: } & \text{subject to } g_1(x) = x_2 + 9x_1 \geq 6 \\ & g_2(x) = -x_2 + 9x_1 \geq 1 \\ & 0.1 \leq x_1 \leq 1 \\ & 0 \leq x_2 \leq 5 \end{aligned}$$

- 已知Pareto-optimal solutions：

區域A： $0.39 \leq x_1^* \leq 0.67 \Rightarrow x_2^* = 6 - 9x_1^*$

區域B： $0.67 \leq x_1^* \leq 1 \Rightarrow x_2^* = 0$

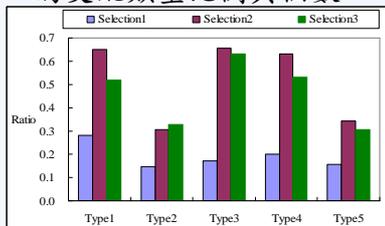


Constr-Ex的實驗設定

- 利用Latin Hypercube Sampling選出十組的起始集合。
- 每次演算流程後產生兩個High-quality Reference Sets與一個Diversity Reference set，每個Reference Set成員各有三個。

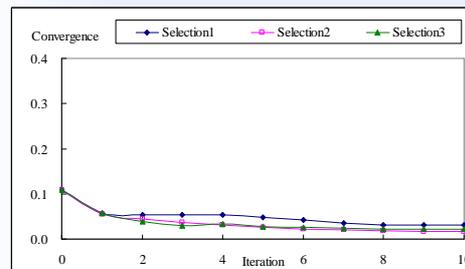
	NSGA-II	本研究
Population Size	40	25
Crossover Rate	0.9	1
Mutation Rate	0	0
Number of Generations	250	10

Constr-Ex：選入非凌駕解集合 的交配類型比例與個數

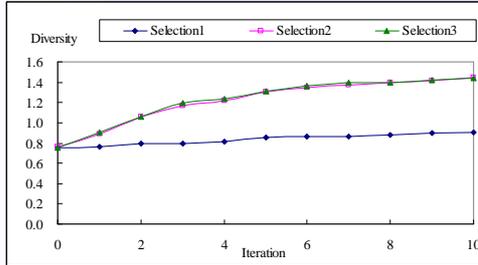


	Type1	Type2	Type3	Type4	Type5
Selection1	3	9	7	8	16
Selection2	7	21	26	27	42
Selection3	5	20	25	21	31

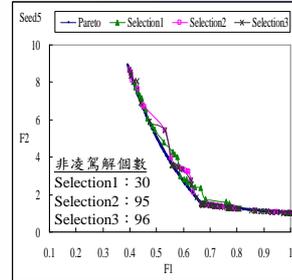
Constr-Ex數學式非凌駕解集合 的收斂情形



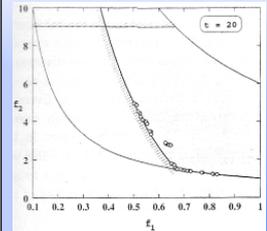
Constr-Ex數學式非凌駕解集合的Diversity情形



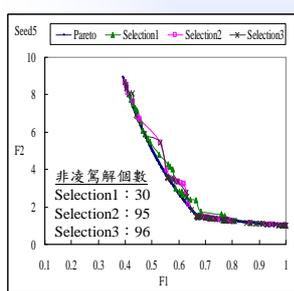
Constr-Ex數學式第五次實驗後的非凌駕解集合



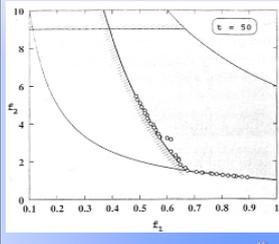
NSGA-II在20次實驗後的非凌駕解集合



Constr-Ex數學式第五組起始解10次實驗後的非凌駕解集合



NSGA-II在50次實驗後的非凌駕解集合



Constr-Ex數學式收斂與多元化指標之平均值與標準差

Algorithm	M_1^+ (Convergence)		Δ (Diversity)	
	\bar{x}	σ_n	\bar{x}	σ_n
NSGA-II	0.0053	0.0003	0.7655	0.0371
Selection1	0.0314	0.0130	0.8985	0.1579
Selection2	0.0164	0.0181	1.4495	0.1086
Selection3	0.0217	0.0252	1.4388	0.1032

數學測試結論

- 起始解建構方式採用LHS可獲得分佈多元化的非凌駕解集合。
- 交配方式同時採用兩點與三點交配可以增加搜尋的範圍。
- 交配類型的選擇比例：交配類型二與交配類型五，因為交配總數過多，造成成功率偏低的情形，但是**應用Diversity Reference Set**，改善了分佈不均勻與找不到端點的現象，如果需要調整每代母體數時，可以減少這兩種類型的交配數量，但是絕對不能忽視Diversity Reference Set對實驗的影響。

數學測試結論

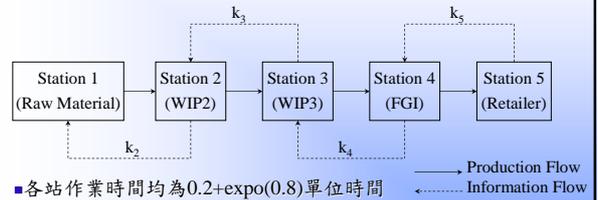
- 非凌駕解集合的收斂速度與解的分佈狀況
 - 收斂情形：應用不同的選擇Reference Sets方式在兩種數學式，發現第五次演算流程後收斂呈現穩定狀態。
 - 多元化狀況：第一種選擇Reference Sets方式的多元化狀況明顯比其他兩種方式好。
- 從Constr-Ex數學式測試發現，本演算法第10次的實驗找出的非凌駕解比NSGA-II第20與50次後的解較多，且分佈範圍廣。

模擬的實驗設定

- 利用Latin Hypercube Sampling選出起始集合。
- 運用第一種選擇Reference Sets的方式：考慮單目標的績效表現。
- 每次演算流程後產生兩個High-quality Reference Sets與一個Diversity Reference set，每個Reference Set成員各有三個。
- Population Size為16個，減半了交配類型二與五的個數。

49

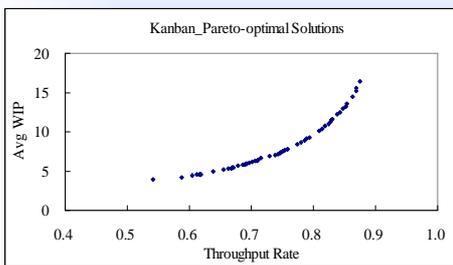
流線型生產系統



- 各站作業時間均為 $0.2 + \text{expo}(0.8)$ 單位時間
- 決策變數： k_i 為第 i 站的庫存水準
- 多目標式：Minimize Average WIP
Maximize Throughput Rate

50

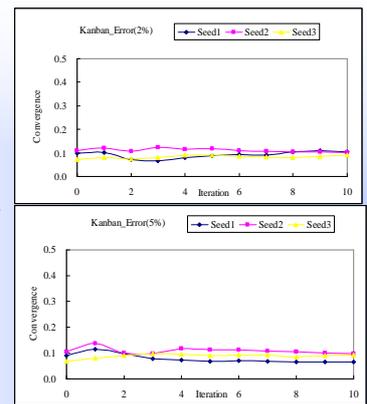
流線型生產系統的 Pareto-optimal Solutions



51

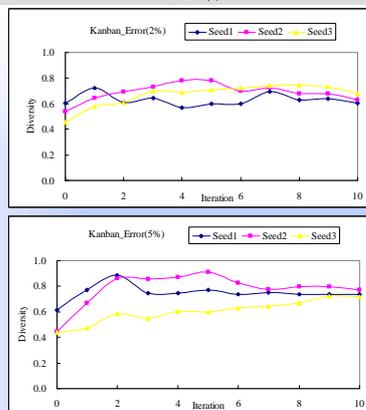
流線型生產系統的收斂情形

- 誤差幅度變大，不影響收斂狀況
- 第四個演算流程後，收斂情形趨於穩定。



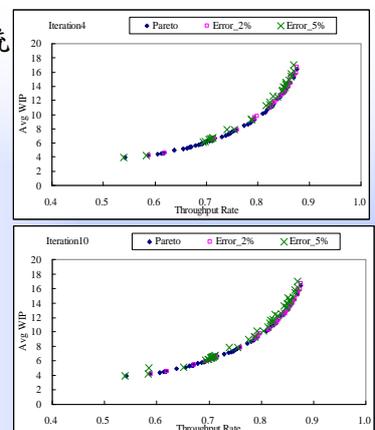
流線型生產系統的多元化狀況

- 誤差幅度變大，無法分辨差異的解變多，多元化指標值變大。



流線型生產系統第二次實驗後非凌駕解集合

- 不同誤差幅度下，實驗都快速收斂
- 沒有找不到端點或某些區段值的情形



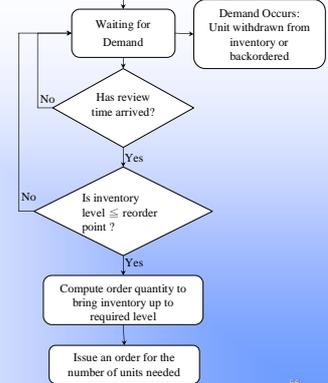
非凌駕解集合中Pareto-optimal Solutions與誤判的個數

		Hit	Type I Error	Type II Error
Error(2%)	Seed1	10	0	30
	Seed2	11	0	29
	Seed3	12	0	32
Error(5%)	Seed1	24	0	18
	Seed2	7	0	38
	Seed3	13	0	26

55

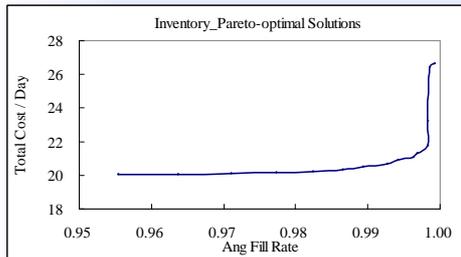
選擇補貨系統

- 決策變數：
Reorder Point (R)
Review Period (P)
Safety Stock (S)
- 多目標式：
Minimize Total Daily Cost
Maximize Fill Rate



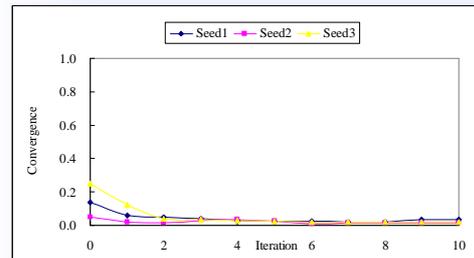
56

選擇補貨系統的 Pareto-optimal solutions



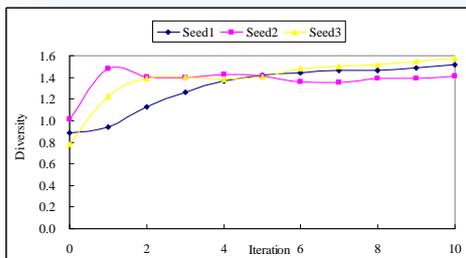
57

選擇補貨系統的收斂情形



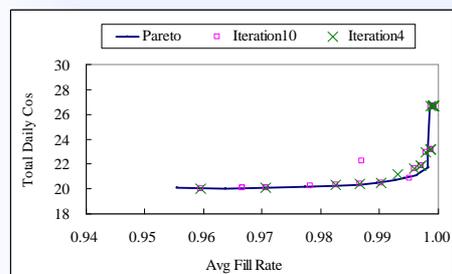
58

選擇補貨系統的多元化狀況



59

選擇補貨系統 第一次實驗後非凌駕解集合



60

非凌駕解集合命中Pareto-optimal Solutions與誤判的個數

		Hit	Type I Error	Type II Error
Error(1%)	Seed1	6	0	10
	Seed2	8	0	5
	Seed3	4	0	11
Error(5%)	Seed1	7	0	12
	Seed2	9	0	11
	Seed3	7	0	17

51

結論

- 本演算法的特色
 - 利用LHS建構起始解系統，
 - 應用High-quality Reference Set與Diversity Reference Set來進行兩點與三點的交配，
 - 非凌駕解更新過程利用NSGA-II的Crowding Distance決定系統設計的去留。

52

結論

- 本演算法應用在不同類型的數學式與模擬問題都可以快速收斂。
- 應用Diversity reference set時，解決了找不出某些端點或分佈不均勻的情形。
- 本演算法在誤差幅度變大時，不影響搜尋非凌駕解的收斂情形。

53

建議

- 加入讓非凌駕解集合增加穩定性的方法，改善解的分佈狀況，保持解的多元化(均勻分佈)。
- 使用突變，增加實驗的範圍與變異性。
- 開發使用者介面，提供給需要利用模擬解決多目標最佳化問題的使用者，有效率的找到所有的Pareto-optimal Solutions。

54



報告結束 敬請指教

Thanks For Your Attention

55