

# 粒子群最佳化巨集啟發式方法求解 多貨艙車輛路線問題之研究

## A PARTICLE SWARM OPTIMIZATION SOLUTION APPROACH FOR THE MULTI-COMPARTMENT VEHICLE ROUTING PROBLEM

韓復華 Anthony F. Han<sup>1</sup>

朱佑旌 Yu-Ching Chu<sup>2</sup>

林致瑄 Jhih-Syuan Lin<sup>3</sup>

(104 年 5 月 26 日收稿，105 年 1 月 12 日第 1 次修改，  
105 年 5 月 25 日第 2 次修改，105 年 6 月 15 日定稿)

### 摘 要

多貨艙車輛路線問題 (Multi-Compartment Vehicle Routing Problem, MCVRP) 是傳統車輛路線問題 (Vehicle Routing Problem, VRP) 的衍生問題之一。在 MCVRP 中每位顧客可有多種物品需要配送，各車輛亦設有多個不同固定容量的隔艙，各自對應一種特定物品的裝載使用。MCVRP 依「不可分送」與「可分送」之條件分為兩種型態，前者要求每位顧客的多種物品必須由單一車輛服務；後者則允許同一顧客由多部車輛分批服務。本研究應用粒子群最佳化 (Particle Swarm Optimization, PSO) 巨集啟發式方法求解 MCVRP。首先，依據「不可分送」與「可分送」的問題型態，分別設計兩種編解碼方法作為粒子解產生與演化學習的基礎。此外，各迭代則採用包括有兩種路線內與六種路線間交換法的變動鄰域下降 (Variable

- 
1. 國立交通大學運輸科技與管理學系教授 (聯絡地址：30010 新竹市東區大學路 1001 號 交通大學運輸科技與管理學系；電話：03-5731680；E-mail: afhan@mail.nctu.edu.tw)。
  2. 國立交通大學運輸科技與管理學系博士候選人。
  3. 國立交通大學運輸科技與管理學系碩士。

Neighborhood Descent, VND) 改善模組以增強搜尋之深度，其中針對「可分送」的問題型態亦提出一個新的  $(1, 0)^*$  鄰域搜尋法。本研究以兩組國際標準例題進行測試，發現 80 題例題中，本研究可求得 16 題現有文獻最佳解，並改善了 34 題文獻最佳解結果。

**關鍵詞：**多貨艙車輛路線問題；粒子群最佳化；變動鄰域下降；巨集啟發式方法

## ABSTRACT

*Multi-compartment vehicle routing problem (MCVRP) is a variant of the conventional vehicle routing problem (VRP). The MCVRP considers multiple products to be delivered, and each product must load on a specific compartment in the vehicle. The problem considers two cases for customer delivery, “no split” and “split”, depending on if the multiple products are allowed to be split among multiple routes. We applied the particle swarm optimization (PSO) metaheuristic approach to solve the MCVRP. Two new solution representation methods were designed to generate and evolve the particles for both the no-split and split cases respectively. A variable neighborhood descent (VND) module with an innovative  $(1, 0)^*$  node-interchange operator was built to improve the quantity of particles during its evolution process. Two sets of benchmark instances for MCVRP were adopted to test the proposed PSO metaheuristic method. Results showed that the PSO method is very competitive as compared to the best algorithms published in the MCVRP literature. Out of the 80 benchmark instances tested, the PSO found 16, and improved 34 best known solutions.*

**Key Words:** Multi-Compartment vehicle routing problem (MCVRP); Particle swarm optimization (PSO); Variable neighborhood descent (VND); Metaheuristic

## 一、前言

近年來，連鎖便利商店已成為民眾生活圈中不可或缺的角色。便利商店提供的物品與服務日趨多元，例如冷凍食品、飲料、民生用品以及服務性物品等等，目標是希望能夠滿足民眾大部分的日常所需。面對此種物品多元化的趨勢，物流配送的作業亦越來越複雜。物流作業進行時，除了將正確的物品依照要求的時間配送至門店以外，更要注意物品有其不同的配送條件以及限制，例如配送車艙的環境條件或是即時性等不同的限制。因此，近年來有許多研究將原本的车辆路線問題 (vehicle routing problem, VRP) 搭配實務上各種不同的配送條件進行調整衍伸出更能夠貼合目前物流作業的服務需求。

多貨艙車輛路線問題 (multi-compartment vehicle routing problem, MCVRP) 即是以 VRP 作為基礎的問題變形。MCVRP 假設各顧客同時具有多種不同特性的物品，因此貨車

上之貨艙分隔為多個固定容量的隔艙，並各自對應一種特定物品的裝載使用。MCVRP 是一個符合實務需求的車輛路線問題，舉凡在生鮮超商低溫配送以及石油產業廢油料回收等配送問題均可應用。圖 1 即為目前在歐洲使用的低溫配送貨車的內部隔艙圖，其中左邊車艙為冷藏貨艙，右邊車艙為冷凍貨艙。

MCVRP 可分為「不可分送」與「可分送」兩種配送條件的問題型態，前者要求每位顧客的多種物品必須由單一車輛服務；後者則允許單一顧客由多部車輛分批服務，兩種不同型態的 MCVRP 目標皆為總路線成本最小。由於 MCVRP 為 VRP 的衍伸問題，在求解的複雜度上同屬 NP-hard，目前相關的解法研究皆是設計以巨集啟發式解法進行求解，期望能在短時間內求得品質佳的近似解。



圖 1 低溫配送貨車內部隔艙圖

資料來源：<http://goo.gl/YUPQVP>

近 20 年來，巨集啟發式解法在 VRP 相關問題上的求解已有非常廣泛的應用，其中群體智慧 (swarm intelligence, SI) 的演算機制亦有相當顯著的績效。群體智慧演算法的特色是藉由大量的個體透過自身求解的經驗提供資訊，進一步幫助群體制定搜尋的策略，以達到最佳化的目標。群體智慧演算法相當繁多，例如基因演算法 (genetic algorithm, GA)、螞蟻最佳化演算法 (ant colony optimization, ACO)、粒子群最佳化演算法 (particle swarm optimization, PSO) 皆曾被有效應用在 VRP 相關問題的求解上。

MCVRP 在實務的應用上是十分具有價值的，由於問題較為新穎，相關的研究並未多見。有鑑於群體智慧在 VRP 相關問題的成功經驗，本研究首次應用粒子群最佳化演算法 (以下簡稱為粒子群演算法或 PSO) 的巨集架構，針對 MCVRP 提出一套新的啟發式解法。本文後續結構如下：第二節為相關文獻回顧；第三節敘述本文應用 PSO 求解 MCVRP 之架構，包括 PSO 粒子群編解碼的設計、變動鄰域下降 (variable neighborhood descent, VND) 改

善模組的建構與 PSO 的執程序序；第四節提出兩組標竿題庫測試的結果與其績效分析的報告；全文以第五節總結。

## 二、文獻回顧

### 2.1 粒子群演算法

粒子群演算法是由 Kennedy 與 Eberhart<sup>[1]</sup> 仿效鳥群覓食的行為所提出。該方法是利用粒子群模擬鳥群在尋找食物的過程中，除了根據本身經驗來判斷最佳覓食地點外，亦藉由特別的訊息傳遞方式相互溝通進行覓食的行為，使粒子群能在解集合的空間內逐步向最佳解移動。

#### 2.1.1 PSO 粒子群演算法

PSO 是一套在多維度連續空間中進行搜尋的演算法，PSO 需要產生  $I$  個粒子進行  $T$  個迭代搜尋。各粒子  $i$  皆代表一個解， $i = 1, 2, \dots, I$ 。粒子  $i$  的編碼  $X_i = \{x_{i1}, x_{i2}, \dots, x_{id}, \dots, x_{iD}\}$ ，其中  $x_{id}$  為粒子  $i$  在維度  $d$  的位置， $d = 1, 2, \dots, D$ 。  $X_i$  能夠在解碼後計算出其相對應的適應值  $\varphi(X_i)$ 。隨著迭代  $t$  的累進， $t = 1, 2, \dots, T$ ，各粒子在求解空間中移動其位置以搜尋更好的結果。各粒子位移的向量會受到以下 3 個因素所影響：(1) 該粒子本身的移動慣性，(2) 本身演算到目前為止所找到的最佳解，即個別最佳解 (personal best, pbest)，以及 (3) 全體粒子演算到目前為止所找到的群體最佳解 (global best, gbest)。利用以上 3 個因子，各個粒子可計算新的移動向量，並決定下一個迭代的位置。粒子移動公式如下，其中式 (1) 為移動速度公式，式 (2) 為粒子移動至新位置的計算公式。

$$v_{id}(t+1) = w \times v_{id}(t) + c_p u_1 \times (P_{id} - x_{id}(t)) + c_g u_2 \times (P_{gd} - x_{id}(t)) \quad (1)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (2)$$

上述兩式中  $v_{id}(t)$  表示第  $t$  個迭代時粒子  $i$  在維度  $d$  的移動向量； $P_{id}$  表示粒子  $i$  的個別最佳解 (pbest) 在維度  $d$  的位置； $P_{gd}$  表示群體最佳解 (gbest) 在維度  $d$  的位置； $c_p$  與  $c_g$  分別為 pbest 與 gbest 的學習因子， $w$  為慣性權重， $u_1$ 、 $u_2$  為介於 0 與 1 間的隨機亂數值， $x_{id}(t)$  表示第  $t$  個迭代時粒子  $i$  在維度  $d$  的位置。

粒子  $i$  從第  $t$  個迭代的位置，即  $x_i(t)$ ，移動至下一迭代的位置，即  $x_i(t+1)$ ，的示意圖如圖 2 所示。

在 PSO 移動公式的參數部分，由於慣性權重值與學習因子會影響粒子在空間搜尋時的範圍以及方向，因此參數設計的好壞會進而影響求解的效果。

關於 PSO 的慣性係數  $w$ ，Shi 與 Eberhart<sup>[2]</sup> 指出當其值越大時，每次的位移幅度相對

較大，對應在求解上則是在全域探索 (global) 的能力會有較佳的表現，亦可避免粒子迅速收斂而陷入區域最佳解；反之，當  $w$  值較小時，則在區域搜尋 (local) 能力較佳，可提升搜尋在深度的效果。因此為使 PSO 在廣度搜尋以及局部搜尋取得平衡，該研究建議慣性係數  $w$  的設定可隨迭代數 ( $t$ ) 線性遞減，慣性係數  $w$  的遞減公式如式 (3) 所示，其中  $T$  表迭代總次數：

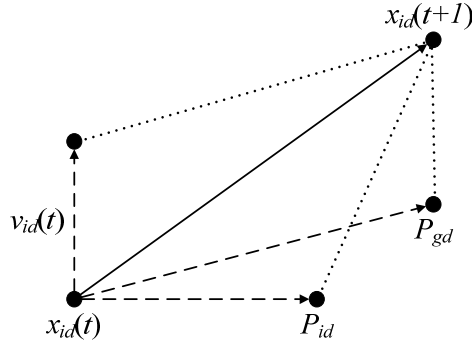


圖 2 粒子移動示意圖

$$w(t) = w(1) - \frac{w(1) - w(T)}{T - 1} \times (t - 1) \quad (3)$$

線性遞減型態的慣性係數  $w$  可使粒子群起先在大範圍進行搜尋，再逐漸由降低粒子的慣性速度，使各粒子進入一個較佳的區域深度搜尋，以獲得品質更好的求解結果。

於學習因子方面， $c_p$ 、 $c_g$  分別代表粒子個體最佳解、群體最佳解影響的比重，當  $c_p$  值較高時，各個粒子能依自身的搜尋經驗有效地搜尋各個可行解區域，但收斂速度較慢；當  $c_g$  值較高時，能使粒子群體迅速移動朝向群體最佳解的區域搜尋，同時意謂粒子搜尋其他空間的可能性減少，易使求解落入局部最佳解。一般設定  $c_p$ 、 $c_g$  為相同值皆為 2 或皆為 1.49445<sup>[3,4]</sup>，意味著粒子在自我搜尋以及在社會模式的搜尋擁有相同的比重。

除了傳統 PSO 的 pbest 與 gbest 兩個學習因子外，Pongchairerks 與 Kachitvichyanukul<sup>[4]</sup> 增加考慮局部最佳解 (local best, lbest) 與鄰近最佳解 (near neighborhood best, nbest) 兩個新的學習因子，提出 GLNPSO 學習架構。其中，lbest 為該粒子個別最佳解 pbest 周圍最靠近的  $K$  個其他粒子中適應值最小的粒子，nbest 則為該粒子周圍適應值與距離比值  $FDR$  (fitness distance ratio) 最大的粒子，對粒子  $i$  而言，其與另一個粒子  $j$  在維度  $d$  上的  $FDR_{jd}$  值定義如下：

$$FDR_{jd} = \frac{\varphi(X_i) - \varphi(P_j)}{|x_{id} - p_{jd}|}, \quad i \neq j \quad (4)$$

式 (4) 中  $\varphi(X_i)$  與  $\varphi(P_j)$  分別為粒子  $i$  現行解與粒子  $j$  的 pbest 之適應值,  $x_{id}$  與  $P_{jd}$  分別為粒子  $i$  在維度  $d$  的位置與粒子  $j$  的 pbest 在維度  $d$  的位置。換言之, 若粒子  $\alpha = \arg \max_{j \in N} \{FDR_{ijd} | i \neq j\}$ , 則粒子  $\alpha$  為粒子  $i$  在維度  $d$  的鄰近最佳解 nbest, 亦即  $P_{\alpha d}$  為  $x_{id}$  的 nbest。GLNPSO 綜合移動慣性與 4 個學習因子 (pbest、gbest、lbest 以及 nbest), 將原先粒子速度更新的公式, 式 (1), 修改為下列之式 (5):

$$v_{id}(t+1) = w(t)v_{id}(t) + c_p u_1 (P_{id} - x_{id}(t)) + c_g u_2 (P_{gd} - x_{id}(t)) \\ + c_l u_3 (P_{id}^L - x_{id}(t)) + c_n u_4 (P_{id}^N - x_{id}(t)) \quad (5)$$

上式中  $P_{id}^L$  與  $P_{id}^N$  分別表示粒子  $i$  的 lbest 與 nbest 在維度  $d$  的位置,  $c_l$  與  $c_n$  分別為 lbest 與 nbest 的學習因子,  $u_1$ 、 $u_2$ 、 $u_3$  以及  $u_4$  為介於 0 到 1 間的隨機亂數值。參數設定方面,  $c_g$ 、 $c_p$ 、 $c_l$  與  $c_n$  在文獻中<sup>[5-7]</sup> 建議以 0.5、0.5、1.5 與 1.5 的組合設定。

### 2.1.2 PSO 求解車輛路線問題

傳統 PSO 的位移公式皆是連續性的計算方式, 基本上可直接適用於連續解空間搜尋的問題<sup>[1,8,9]</sup>。若要將 PSO 應用於離散的 VRP 上, 則需要特殊的解碼技巧將粒子的編碼轉換成二元整數型態的路線解。近年來, 在 PSO 應用於 VRP 相關問題的求解上已有大量的研究, 並且產生許多良好的成果。

PSO 最早應用於 VRP 的研究為 2006 年的 Chen 等人<sup>[10]</sup>, 作者將離散型的 PSO 編碼搭配模擬退火法 (simulated annealing) 求解傳統 CVRP (capacitated VRP), 測試 16 題 VRP 標竿例題, 結果與已知最佳解 (best known solution, BKS) 平均誤差為 0.97%。

Ai 與 Kachitvichyanukul<sup>[5,11]</sup> 提出 SR-1 與 SR-2 兩種連續性編解碼模式搭配 GLNPSO 學習策略應用於求解 CVRP 以及同時收送貨的 VRPSPD (VRP with simultaneous pickup and delivery)。求解的結果與當時 BKS 的平均誤差在 Christofides 等人<sup>[12]</sup> 提出的 14 題 CVRP 標竿例題部分為 0.88%, 另在 98 題 VRPSPD 標竿例題部分則為 -4.21%。

Marinakis 等人<sup>[13]</sup> 年將 PSO 混合 3 種方法論求解 CVRP 問題, 包含多重鄰域搜尋—貪心隨機適應搜尋程序 (multiple phase neighborhood search—greedy randomized adaptive search procedure)、擴大鄰域搜尋 (expanding neighborhood search) 以及路徑重新連結 (path relinking)。該研究以 14 題 CVRP 標竿例題<sup>[8]</sup> 進行測試, 結果與 BKS 誤差僅 0.08%。

Qi<sup>[14]</sup> 嘗試以離散編碼 PSO 配合 ILS (iterated local search) 測試 CVRP, 該研究於 12 題 CVRP 標竿例題, 平均誤差為 0.57%。

Goksal 等人<sup>[15]</sup> 提出一套 PSO 搭配變動鄰域尋優法 (variable neighborhood descent, VND) 求解 VRPSPD 與 VRPMPD (VRP with mixed pickup and delivery) 兩個 VRP 相關問題, 求解結果與當時 BKS 的平均誤差分別為 0.01% (VRPSPD) 以及 -1.58% (VRPMPD)。

韓復華與楊禮瑛<sup>[6]</sup> 參考 SR-2 的編碼方式<sup>[5]</sup> 以及 GLSPSO 的學習策略求解 OVRP。

該研究以 2-opt\*、1-0、1-1、2-opt 以及 Or-opt 等 5 種鄰域改善法搭配求解。測試 OVRP 標竿題庫整體平均誤差僅為 0.57%。該研究亦嘗試求解 VRP，整體績效平均誤差率為 0.51%。

Belmecheri 等人<sup>[16]</sup>嘗試將 PSO 應用於求解具時窗限制的多車種混合回程取貨車輛路線問題 HVRPMBTW (VRP with heterogeneous fleet, mixed backhauls, and time windows)，結果發現 PSO 的求解績效優於 ACO。

Norouzi 等人<sup>[17]</sup>提出具競爭時窗週期性車輛路線問題 PVRPCTW (periodic vehicle routing problem with competitive time windows)，並利用粒子具重新產生機制的 IPSO 進行求解測試。在小型例題測試部分先與 Lingo 比較，發現 IPSO 在求解時間上限 1 小時的條件下，與最佳解的求解誤差為 0.01%，優於 Lingo 的 0.04%。在大型例題部分則以 IPSO 與傳統 PSO 進行測試比較，求解結果呈現 IPSO 在所有例題皆可求得優於 PSO 的答案，求解時間約為傳統 PSO 的 1.5 倍。

Kachitvichyanukul 等人<sup>[7]</sup>提出一般化多場站多重取送需求車輛路線問題 GVRP-MDMPDR (generalized multi-depot vehicle routing problem with multiple pickup and delivery requests)，並以 GLNPSO 搭配 3 種不同的粒子編碼進行求解測試。

## 2.2 MCVRP 數學定義以及相關研究

### 2.2.1 MCVRP 數學模式

MCVRP 由 Fallahi 等人<sup>[18]</sup>首次提出，並定義問題的網路為完全連接且無方向性，其中節點集合  $N' = \{0, 1, \dots, n\}$  (節點 0 表示場站) 以及  $n$  個顧客的顧客集合  $N = \{1, 2, \dots, n\}$ ，各節線  $(i, j)$  的距離成本  $c_{ij} = c_{ji}$ 。假設每個顧客有  $p$  種不同的分艙物品需要配送，令  $P = \{1, 2, \dots, p\}$  代表物品集合。所有的需求由  $v$  輛相同規格的車輛進行服務 (單一車種型態)。每輛車輛具有  $p$  個貨艙，車隊的集合  $V = \{1, 2, \dots, v\}$ 。各車輛的第  $p$  個貨艙僅能裝載物品  $p$ ，各貨艙的容量  $Q_p$  固定已知。顧客  $i$  對於物品  $p$  的需求  $q_{ip}$  大於等於 0，且小於等於  $Q_p$ ，在不同物品間的需求量無特定規則。各車輛具有服務時間長度的限制  $L$ 。MCVRP 的數學模式如下所述：

$$\text{Min} \quad \sum_{i,j \in N} \sum_{k \in V} c_{ij} x_{ijk} \quad (6)$$

$$\text{s.t.} \quad \sum_{i \in N'} x_{ijk} \leq 1, \quad \forall j \in N, \forall k \in V, \quad (7)$$

$$\sum_{i \in N'} x_{ijk} = \sum_{i \in N'} x_{jik}, \quad \forall j \in N, \forall k \in V \quad (8)$$

$$\sum_{i,j \in N'} x_{ijk} \leq |S| - 1, \quad \forall k \in V, \forall S \subseteq V, |S| \geq 2, \quad (9)$$

$$y_{jkp} \leq \sum_{i \in N'} x_{ijk}, \quad \forall j \in N, \forall k \in V, \forall p \in P, \quad (10)$$

$$\sum_{k \in V} y_{jkp} = 1, \quad \forall j \in N, \forall p \in P, \quad (11)$$

$$\sum_{j \in N} y_{jkp} q_{jp} \leq Q_p, \quad \forall k \in V, \forall p \in P, \quad (12)$$

$$\sum_{i, j \in N'} c_{ij} x_{ijk} \leq L, \quad \forall k \in V, \quad (13)$$

$$x_{ijk} \in \{0, 1\}, \quad \forall i \in N', \forall j \in N', i \neq j, \forall k \in V, \quad (14)$$

$$y_{jkp} \in \{0, 1\}, \quad \forall j \in N, \forall k \in V, \forall p \in P, q_{jp} \neq 0, \quad (15)$$

上述數學模式中，式 (6) 為目標函數，表示車輛行駛距離總成本最小化；式 (7) 表示限制每一顧客點僅能由一輛車完成服務；式 (8) 表示流量守恆；式 (9) 避免子迴路產生；式 (10) 表示當車輛  $k$  配送商品  $p$  給顧客  $j$  時，設定變數  $y_{jkp}$  值為 1，否則為 0；式 (11) 表示顧客  $j$  對於商品  $p$  的需求由一輛車服務；式 (12) 與式 (13) 分別表示車容量限制與車輛時間限制；式 (14) 表示決策變數  $x_{ijk}$  為二元變數，當車輛  $k$  自節點  $i$  行駛至節點  $j$  即為 1，否則為 0；式 (15) 表示決策變數  $y_{jkp}$  為二元變數，當顧客  $j$  由車輛  $k$  服務產品  $p$  即為 1，否則為 0。

## 2.2.2 MCVRP 相關解法回顧

多艙等特性的車輛路線相關問題最早在 1979 年被 Christofides 等人<sup>[12]</sup>提及，並認為是 VRP 的延伸問題。後續則開始有相關的研究產生，例如針對食品雜貨<sup>[18,19]</sup>，石油配送<sup>[20-22]</sup>或是資源回收<sup>[23]</sup>的實務應用。雖然這些研究皆具有多艙等的特性，但所針對的個案情境不同，研究所設定的限制亦有差異。本研究所探討的 MCVRP 問題為 2008 年 Fallahi 等人<sup>[18]</sup>所提出的研究問題。該研究針對 MCVRP 提出數學模式，明確定義 MCVRP 的問題目標與限制。研究中並提出測試的標竿題庫，以供後續研究參考。目前針對 MCVRP 提出解法並以標竿題庫進行測試的研究還未多見。以下分別進行回顧，並作為後續在本研究求解績效比較的對象。

Fallahi 等人<sup>[18]</sup>設計以瀰母演算法 (memetic algorithm, MA) 以及禁制搜尋法 (tabu search, TS) 兩套巨集啟發式解法分別搭配 2-opt、Relocate 和 1-Interchange 求解 MCVRP。作者提出 2 套共 40 題例題的標竿題庫搭配進行測試，並以「不可分送」以及「可分送」兩種問題型態進行 MA 與 TS 兩種方法的績效比較。測試的結果發現，整體上 TS 的求解精確度優於 MA，求解效率則是 MA 優於 TS。「可分送」的平均求解績效平均優於「不可分送」。

Muyldermans 等人<sup>[24]</sup>提出 guided local search (GLS) 求解 MCVRP。該研究使用節省法建構起始解並搭配 2-opt、Relocate、Exchange 以及 Cross 等 4 種鄰域搜尋法進行起始解改善，隨後進入 GLS 流程直到預設的停止條件達到為止。該研究以第一組標竿題庫中「無時間限制」之 9 題例題搭配「可分送」的問題型態進行績效測試，發現求得 5 題優於原先



Fallahi 等人<sup>[18]</sup>的求解結果。

Mendoza 等人<sup>[25]</sup>考慮隨機性需求下的 MCVRPD (MCVRP with stochastic demands) 進行求解。該研究以 MA 作為主要求解架構，並搭配 Relocate 以及 2-opt 鄰域搜尋法對染色體進行優化改善。該研究以不可分送的配送限制測試了 40 題標竿題庫，結果突破 13 題。

## 2.3 文獻回顧一小節

本研究回顧 PSO 相關演進以及應用於 VRP 相關之研究，發現該法在 VRP 問題上具有良好的求解效果。在 MCVRP 方面，發現多貨艙的車輛路線問題已在實務上非常具有應用的潛力，但直到 2008 年才由 Fallahi 等人<sup>[18]</sup>提出問題之數學模式，因此目前相關研究尚未多見。

因此，本研究擬以 GLNPSO 學習策略應用於求解 MCVRP。參考 Ai 與 Kachitvichyanukul<sup>[5, 11]</sup>提出的 SR-1 編解碼模組，設計適合 MCVRP 應用的粒子解產生模組。並針對標竿題庫不同艙「可分送」的特性，設計有效的鄰域交換法以增加搜尋深度。

## 三、應用 PSO 求解 MCVRP 之架構設計

在本研究所設計的 PSO 求解架構中是以 GLNPSO 作為學習策略，粒子群中的各粒子皆代表一個獨立的 MCVRP 的可行解，粒子數為粒子群的規模。各粒子可依其編碼的多維度變數值，對應到一個位置向量。本研究提出適當的解碼機制，將連續性的粒子編碼轉換成離散性的 MCVRP 粒子解。在解碼的機制中，我們將 MCVRP 的限制加入，使得解碼後的粒子解是為一個 MCVRP 的可行解。我們利用變動鄰域下降法改善每個粒子解，並計算改善後的解目標值作為該粒子的適應值。各粒子的適應值將決定出該迭代的學習對象，包含 GLNPSO 中的全域最佳解 gbest、個體最佳解 pbest、局部最佳解 lbest 以及鄰近最佳解 nbest。後續各粒子利用 GLNPSO 學習公式決定移動的速度並更新其編碼（位置向量）。本研究的求解架構即是反覆進行上述的解碼，改善以及學習更新等 3 個步驟，最後輸出過程中搜尋品質最佳的 MCVRP 路線結果。

本研究擬應用 GLNPSO 學習策略求解 MCVRP，分別設計 MCSR-1 以及 MCSR-2 兩套編解碼模組求解「不可分送」與「可分送」兩種類型的 MCVRP 問題。優化部分，除了傳統路線內與路線間的鄰域交換法以外，亦考慮多艙配送的問題特性，調整傳統 VRP 之鄰域搜尋法以強化求解功能，提出針對「可分送」問題特性的 (1, 0)\*交換法以增加求解深度。本研究的整體求解架構主要包括 3 個部分：粒子群編解碼機制，鄰域搜尋模組以及 PSO 巨集求解程序，分別敘述於以下 3 個小節。

### 3.1 粒子群編解碼機制

Fallahi 等人<sup>[18]</sup>將 MCVRP 分為「不可分送」與「可分送」兩種問題型態，「不可分送」限制顧客的所有物品僅能由單一車輛進行服務；「可分送」則允許物品依不同的特性可分開由不同的車輛進行服務。本研究針對這兩種不同的配送型態分別提出兩套新的 MCSR-1 與 MCSR-2 編解碼模組。

良好的編解碼方法是 PSO 在求解屬於非連續性的 VRP 相關問題的關鍵，因此我們參考 Ai 與 Kachitvichyanukul<sup>[5,11]</sup>所提出的 SR-1 的編解碼概念，加入多艙等特性後，設計 MCSR-1 與 MCSR-2。SR-1 編解碼方式簡述如下：SR-1 的編碼是 1 個  $n+2m$  維度的連續性編碼 ( $n$  為顧客數； $m$  為車輛數)，其中  $2m$  代表  $m$  部車輛於兩度空間座標  $(x, y)$  的虛擬位置。解碼時，先利用前  $n$  維度的位置由小而大排列產生顧客的清單，再以各顧客距各車輛虛擬點座標的遠近產生選用各車輛服務的優先次序，最後以最省插入法構建出對應的粒子解。然而，SR-1 編碼利用  $(x, y)$  座標建立車輛的虛擬點以構建車輛配送的種子路線的作法，是以兩點間直線距離的歐基里德兩度空間為前提假設。此種設計在車輛必須行駛在街道路網上的現實世界中是不切實際的。有鑑於此，本研究提出的 MCSR-1 與 MCSR-2 編解碼模組移除了原 SR-1 編碼的  $(x, y)$  座標位置設計，改以實際顧客點直接作為車輛種子點產生種子路線。使 MCSR-1 與 MCSR-2 在解碼時僅需要點與點間的距離矩陣即可應用，如此更能夠符合實務的需求。以下分別說明兩種編解碼模組的編碼方式以及解碼程序。

#### 3.1.1 MCSR-1 編解碼模組

MCSR-1 是考慮「不可分送」的配送條件下所設計的編碼方式，其維度 ( $D$ ) 等於顧客數 ( $n$ )。粒子群的初始化編碼是將各粒子的所有維度值以 0 到 1 間的隨機亂數產生完成。各粒子編碼後，便可進行後續解碼以及迭代學習的步驟。

MCSR-1 把一個  $X_i$  編碼轉換成一個 MCVRP 車輛路線解的「解碼」過程包括以下步驟：

步驟 1. (編碼排序)：將  $X_i$  編碼內的各個維度值  $x_{id}$  由小而大進行排序，因總維度  $D$  等於顧客數  $n$ ，排序後的序列即定義為待服務顧客的集合  $C$ 。

步驟 2. (選擇種子點並產生起始路線)：

2-1：以下列式 (16) 與式 (17) 分別計算預估車輛使用數  $m$ ，以及車輛平均服務顧客數  $g$ 。

$$m = \left\lceil \sum_{i=1}^N \sum_{p=1}^P q_{ip} / \sum_{p=1}^P Q_p \right\rceil + 1 \quad (16)$$

$$g = \lceil D/m \rceil - 1 \quad (17)$$

2-2：設定各起始路線對應之種子點的候選集合  $L = C$ ，路線種子點集合  $S = \phi$ 。

2-3：選擇  $L$  中第 1 順位的顧客  $f$  為路線種子，構建一條起始路線  $\{0-f-0\}$ ，將  $f$  移出

集合  $L$  並加入集合  $S$ ，即  $L = L \setminus \{f\}$  且  $S = S \cup \{f\}$ 。

2-4：若  $|S| = m$ ，即已產生足夠之起始路線，則將所有的車輛種子點由  $C$  中移出，更新待服務顧客集合  $C = C \setminus S$ ，前往步驟 3，否則至步驟 2-5。

2-5：為避免種子路線彼此過於接近，自候選集合  $L$  中移出與  $f$  點距離成本最小的前  $g$  個顧客，返回步驟 2-3。

步驟 3. (插入路線的順位排序): 對  $C$  中的各顧客點，依其與各路線種子顧客的距離，由近而遠建立其插入路線的優先次序。

步驟 4. (將顧客依序插入或新創路線): 選擇  $C$  中第 1 順位的顧客為  $c$ ，依照前一步驟產生的優先路線次序，嘗試在車輛裝載可行的前提下，以最省插入法將  $c$  插入路線成本增加最少的位置；若優先路線無法完全服務顧客  $c$  所有物品的需求量，則考慮插入次優先路線。此時有可能發生，所有現行路線均無可行的車容量可服務  $c$ ，則新創一條路線  $\{0 - c - 0\}$ ，並將其設為其他尚未服務之顧客待插入路線的最後順位。待  $c$  點完成插入或新構路線後，則從  $C$  中移出，即  $C = C \setminus \{c\}$ 。

步驟 5. (停止條件): 若  $C = \phi$ ，則停止，並輸出粒子  $i$  的路線解  $S_i$ ；否則回到步驟 4。

圖 3 為 MCSR-1 編解碼的示意簡例，假設顧客數  $n = 6$ ，粒子  $i$  的編碼  $X_i$  的維度  $D = n = 6$ ，首先以隨機亂數產生 6 個維度值，如圖 3a 示，即完成此粒子的編碼。解碼階段的步驟 1：依  $x_{id}$  由小而大產生待服務顧客集合  $C = \{6, 4, 1, 3, 5, 2\}$ ，如圖 3b 所示。步驟 2：計算預估車輛數量  $m$  以及車平均服務顧客數  $g$ ，本例假設  $m = 2$ ，則  $g = \lceil 6/2 \rceil - 1 = 2$ 。設定車輛種子點的候選集合  $L = C = \{6, 4, 1, 3, 5, 2\}$ ，車輛種子點集合  $S = \phi$ 。選擇  $L$  中第 1 個顧客，即點 6，為第一條起始路線的種子點，建立第 1 條起始路線  $\{0 - 6 - 0\}$ ，並設定  $L = L \setminus \{6\}$  且  $S = S \cup \{6\}$ 。此時  $|S| = 1 \neq m$  (2)，則自集合  $L$  移除點 6 以及距離點 6 最近的 2 個顧客 ( $g = 2$ )，茲假設點 4 與點 5 是  $L$  集中與點 6 距離成本最小的兩個顧客，則更新  $L = L \setminus \{5, 4\} = \{1, 3, 2\}$ 。再由  $L$  中選出第 2 條起始路線的種子點點 1，建立第 2 條起始路線  $\{0 - 1 - 0\}$ ，並更新  $L = L \setminus \{1\}$  且  $S = S \cup \{1\}$ 。此時  $|S| = 2 = m$ ，設定  $C = C \setminus S = \{4, 3, 5, 2\}$ ，如圖 3c 所示。步驟 3：分別對  $C$  中的顧客，依照與起始路線種子點的距離，由近而遠產生插入路線優先順序，如圖 3d 所示。步驟 4：依序對  $C$  中的顧客以最省插入法插入優先次序最高且可行的種子路線中成本增加最少的位置。若所有的路線皆無可行的剩餘車容量以服務目標顧客，則新增路線服務之，假設圖 3e 在插入點 2 時已無可行路線，則新增第 3 條路線  $\{0 - 2 - 0\}$  服務之。若還有尚未服務之顧客，則將路線 3 設為其他尚未服務之顧客點的最後插入路線。各需求點完成插入後，於  $C$  中移除，反覆執行至  $C = \phi$  時停止解碼，如圖 3e 所示，並輸出最後的結果為  $X_i$  編碼對應的路線解  $S_i$ ，如圖 3f 所示。

### 3.1.2 MCSR-2 編解碼模組

本研究另針對「可分送」的配送條件，將前述 MCSR-1 調整修正為 MCSR-2 以求解對

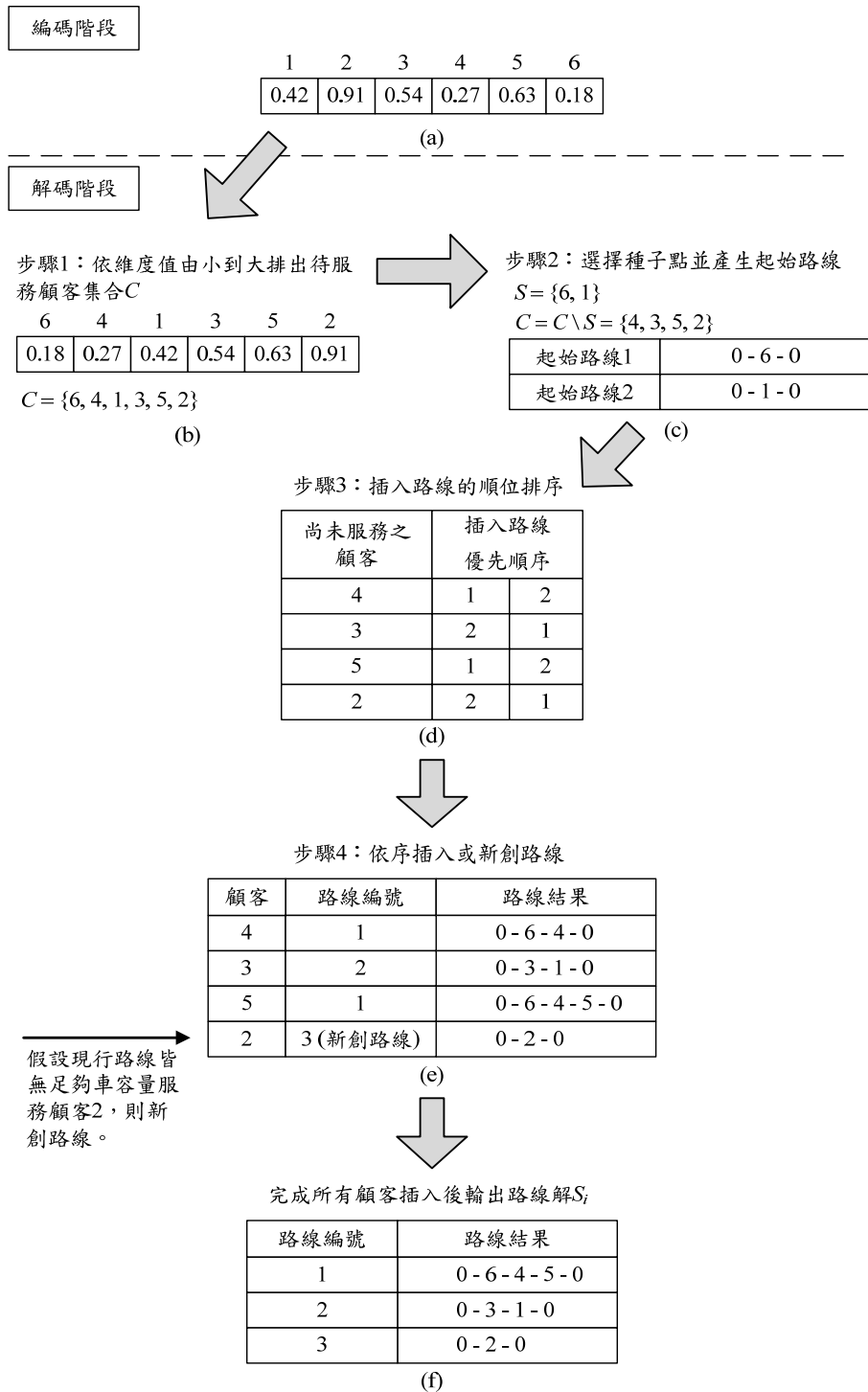


圖 3 MCSR-1 粒子編解碼示意簡例

應的 MCVRP。由於在可分送的考慮下，單獨一個顧客可具有  $p$  個不同物品的需求，故將各顧客對應的每個物品拆解成一個獨立的需求點，以進行編碼。換言之， $n$  個顧客就考慮  $n \times p$  個配送需求點。因此，MCSR-2 粒子群編碼的總維度  $D = np$ ，起始的編碼依然是對每個粒子各個維度值以隨機亂數 (0 至 1) 產生。

MCSR-2 編解碼的步驟程序，基本上與 MCSR-1 相同，為避免重複的敘述，茲以圖 4 的簡例來說明 MCSR-2 的步驟。該例有 6 個顧客 ( $n = 6$ )，兩種可分送的物品 ( $p = 2$ )，故其粒子編碼之維度  $D = 6 \times 2 = 12$ 。圖 4a 表示 1 個隨機產生的具有 12 個維度值的粒子編碼，其中第  $(2k-1)$  與第  $2k$  個的維度值分別代表第  $k$  個顧客 ( $k = 1, 2, \dots, n$ ) 第 1 種 (a 類) 與第 2 種 (b 類) 物品的服務需求點。換言之，該編碼  $X_i$  的第 9 與第 10 位的編碼值，0.57 與 0.15 即分別對應第 5 個顧客對 a 與 b 兩類物品的需求點。

圖 4 解碼階段的步驟 1：依照  $x_{id}$  由小而大產生待服務顧客集合  $C = \{5b, 3b, 1b, 4b, 6b, 1a, 2b, 5a, 3a, 4a, 6a, 2a\}$ ，如圖 4b 所示。步驟 2：計算預估車輛數量  $m$  以及車平均服務顧客數  $g$ ，本例暫假設  $m = 2$  則  $g = \lceil 12/2 \rceil - 1 = 5$ 。設定起始路線各種子點的候選集合  $L = C = \{5b, 3b, 1b, 4b, 6b, 1a, 2b, 5a, 3a, 4a, 6a, 2a\}$ ，車輛種子點集合  $S = \phi$ 。選擇  $L$  中第 1 個顧客，即點 5b，為第 1 條起始路線的種子點，建立第 1 條起始路線  $\{0 - 5b - 0\}$ ，設定  $L = L \setminus \{5b\}$  且  $S = S \cup \{5b\}$ 。此時  $|S| = 1 \neq m$  (2)，則自集合  $L$  移除點 5b 以及距離點 5b 最近的 5 個顧客 ( $g = 5$ )，茲假設點 4a、4b、5a、6a 與點 6b 是  $L$  集中與點 5b 距離成本最小的 5 個顧客，則將其由  $L$  中移出，並更新  $L$  為  $\{3b, 1b, 1a, 2b, 3a, 2a\}$ 。再由  $L$  中選出第 2 條起始路線的種子點顧客 3b，建立第 2 條起始路線  $\{0 - 3b - 0\}$ ，並更新  $L = L \setminus \{3b\}$  且  $S = S \cup \{3b\} = \{5b, 3b\}$ 。此時  $|S| = 2 = m$ ，設定  $C = C \setminus S = \{1b, 4b, 6b, 1a, 2b, 5a, 3a, 4a, 6a, 2a\}$ ，如圖 4c 所示。步驟 3：分別對  $C$  中的顧客，依照與起始路線種子點的距離，由近而遠產生插入路線優先順序，如圖 4d 所示。步驟 4：依序對  $C$  中的顧客以最省插入法插入優先次序最高且可行的種子路線中。圖 4e 在插入點 2a 時，原最優先的路線為路線 2，假設路線 2 該時容量已不足，故插入次優先的路線 1。各需求點完成插入後，於  $C$  中移除，反覆執行至  $C = \phi$  時停止解碼，如圖 4e 所示，並輸出最後的結果為  $X_i$  編碼對應的路線解  $S_i$ ，如圖 4f 所示。

### 3.2 變動鄰域下降改善模組

所有的粒子編碼  $X_i$  解碼轉換為路線解  $S_i$  後，皆以變動鄰域下降 (variable neighborhood descent, VND) <sup>[26]</sup> 改善模組搜尋其局域最佳解 (local optimal) 進行改善。為提高求解的精確度，VND 改善模組執行時會利用 8 種交換法進行深度搜尋改善，包含兩種路線內以及 6 種路線間交換法。路線內交換法部分，包括 2-opt 節線交換法 <sup>[27]</sup> 及 Or-opt 節點交換法 <sup>[28]</sup>。圖 5 為兩種路線內交換法的示意圖，圖 5a 為交換改善前的路線，圖 5b 與 5c 分別是執行 2-opt 與 Or-opt 交換法後之示意圖。

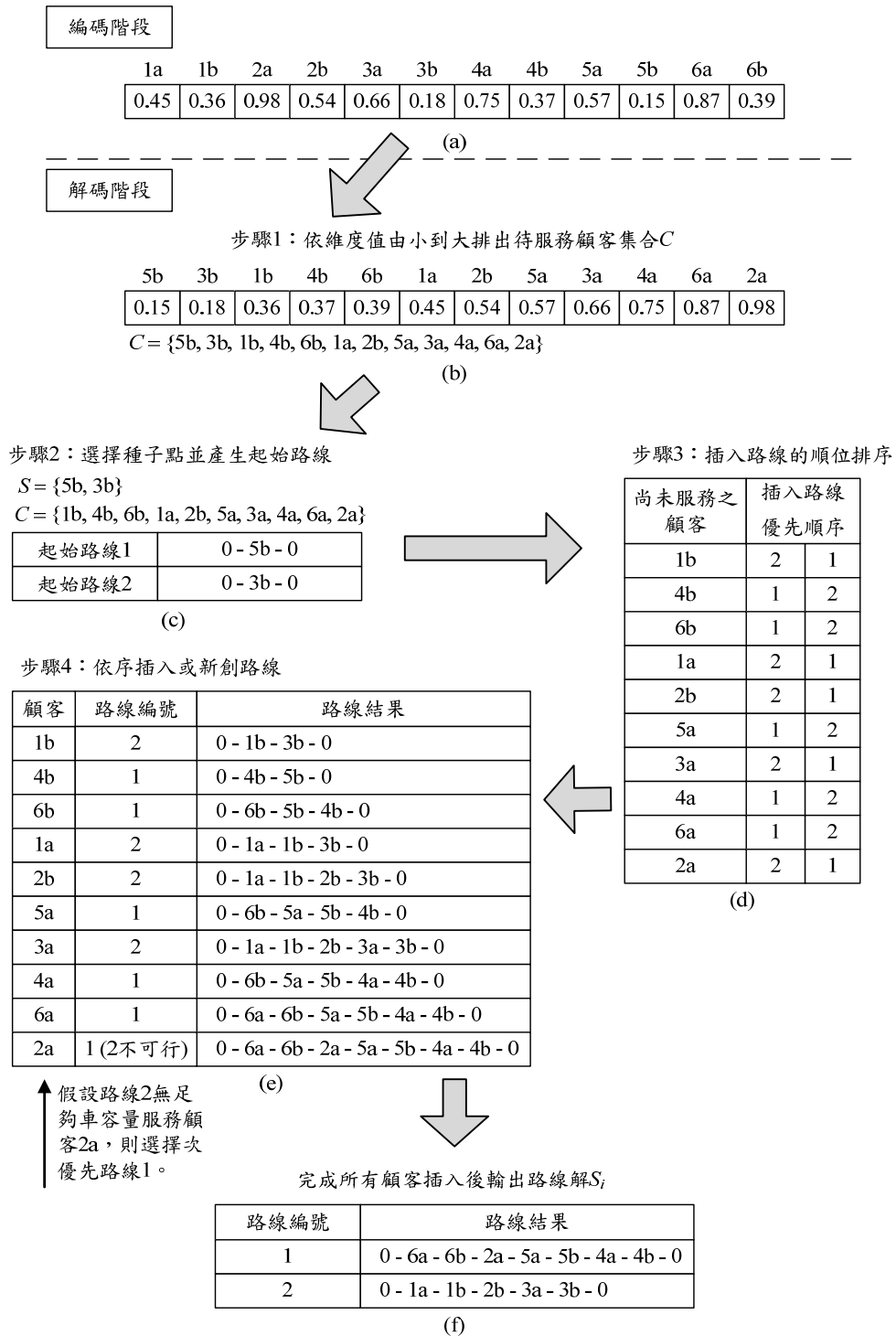


圖4 MCSR-2 粒子編解碼示意圖

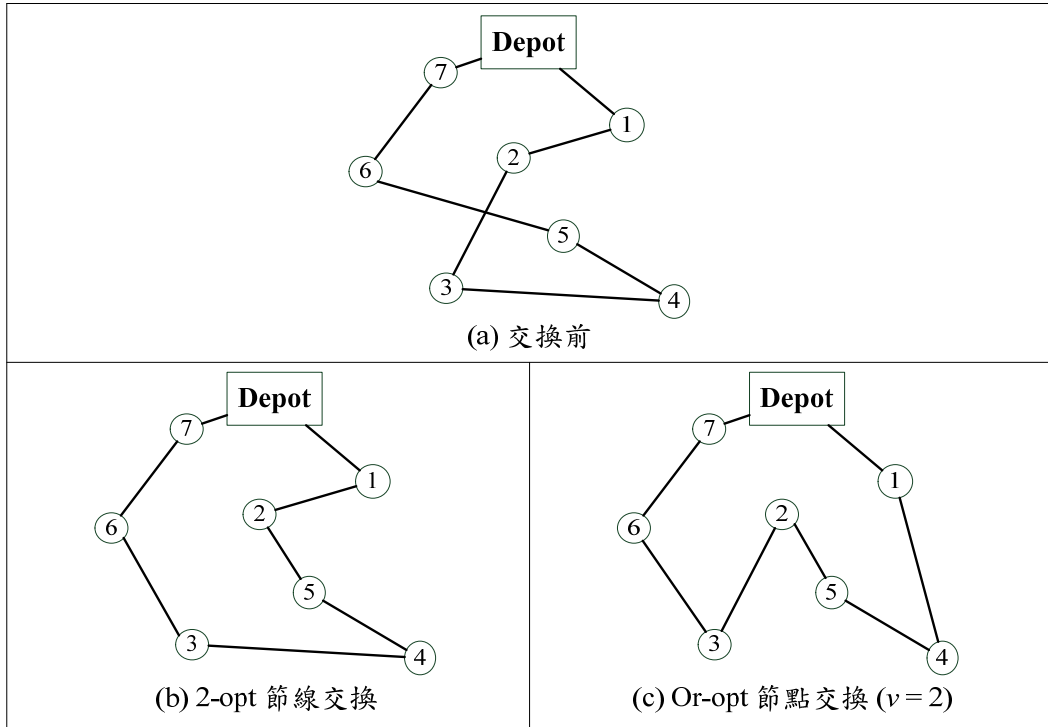


圖 5 路線內交換法示意圖

具體而言，圖 5b 表示將原圖 5a 路線的節線 (2, 3) 以及 (5, 6) 移除後，重新連結節線 (2, 5) 以及 (3, 6) 形成的改善路線。Or-opt 交換是將原路線連續  $v$  個節點 ( $v=1, 2, 3$ ) 移動至路線內其他位置以尋求改善。圖 5c 是  $v=2$  時 Or-opt 交換後的一個示意，其中原路線 (圖 5a) 的兩個節點 2、3 及節線 (2, 3) 被移動至節點 5 與 6 之間。

本研究另採用 6 種路線間交換法，包括 5 種  $\lambda$ -節點交換法<sup>[29]</sup> 以及 2-opt\*節線交換法<sup>[30]</sup>。 $\lambda$ -節點交換法是在兩條路線  $R1$  與  $R2$  間，分別選取至多  $\lambda$  個節點互相移動至另一條路線上的各種交換改善組合。本研究採用 Cordeau 與 Laporte<sup>[31]</sup> 之建議，將  $\lambda$  設為 2。在此設定下所使用的  $\lambda$ -節點交換法在排除對稱重複的組合之後，包含 (1, 0)、(1, 1)、(2, 0)、(2, 1) 以及 (2, 2) 等 5 種不同的節點交換 (移動) 法。圖 6 為 6 種路線間交換法的示意圖，其中圖 6a 為交換前的路線示意圖，圖 6b 至 6f 分別為 5 種  $\lambda$ -節點交換法搜尋後的路線示意圖，圖 6g 則為 2-opt\*節線交換法搜尋後的路線示意圖。以圖 6e 為例，其代表圖 6a 兩條路線， $R1$  與  $R2$ ，經過 (2, 1) 節點交換的示意。具體而言，圖 6e 是將原先  $R2$  的兩個節點 (節點 10 與 11) 與  $R1$  上的一個節點 (節點 1) 進行交換的結果。圖 6g 的 2-opt\*節線交換，則是分別移除原  $R1$  與  $R2$  路線的 (1, 2) 與 (9, 10) 節線，再分別以 (1, 10) 與 (9, 2) 節線予以連接的結果。

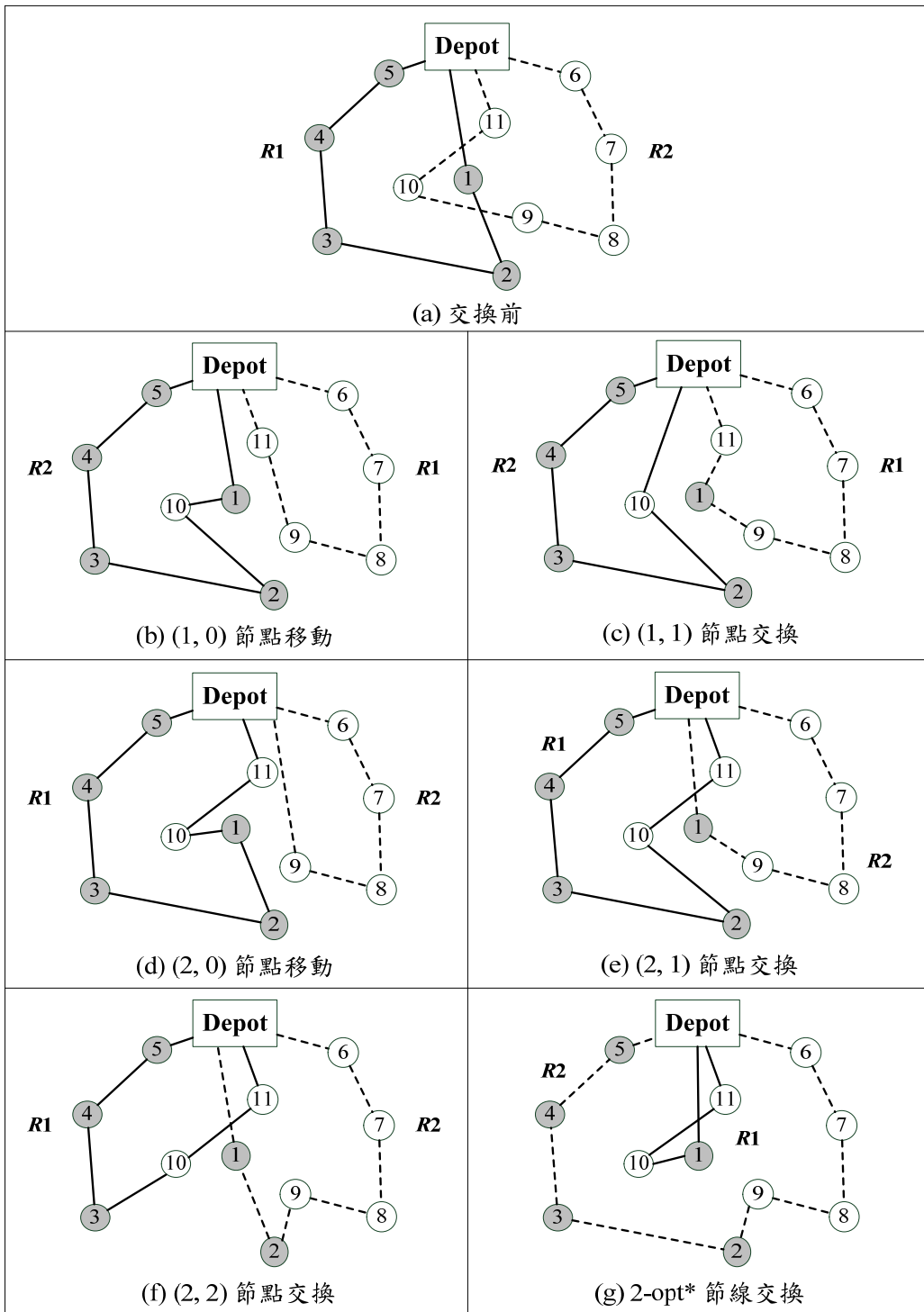


圖 6 路線間交換法示意圖



由於「可分送」配送型態在 MCSR-2 解碼時是將各物品需求視為獨立的需求點，本研究另提出一套針對「可分送」配送型態特性的路線間節點交換法，稱之為  $(1, 0)^*$  節點移動法。 $(1, 0)^*$  為  $(1, 0)$  節點移動的延伸，是一個具適應性且搜尋廣度更強的交換方式。傳統的  $(1, 0)$  節點移動，僅考慮將某路線上的一個節點移除再插入至另一條路線的交換改善法。但由於可分送型態的每個顧客都有  $p$  種不同的物品需求，本研究新設計的  $(1, 0)^*$  交換法即考慮到當顧客  $i$  從某單一路線移除時，其原路線上的  $p$  種物品可以分別插入到其他  $r$  ( $r \leq p$ ) 條不同路線的各種可能性，並選取其中最優者為交換改善的結果。以圖 7 為例，假設  $p = 2$ ，顧客  $i$  的兩個物品  $ia$  與  $ib$  同時由路線  $R2$  服務，如圖 7a 所示。 $(1, 0)^*$  會同時考慮以下交換方式，並選擇改善較多者執行：(1)  $r = 1$  的情況，如圖 7b 所示，將需求  $ia$  與  $ib$  同時移動至路線  $R1$ ，此亦為傳統的  $(1, 0)$  節點移動；(2)  $r = 2 = p$  的情況，如圖 7c 所示，將物品  $ia$  與物品  $ib$  分別移動到  $R1$  與  $R3$  兩條路線。

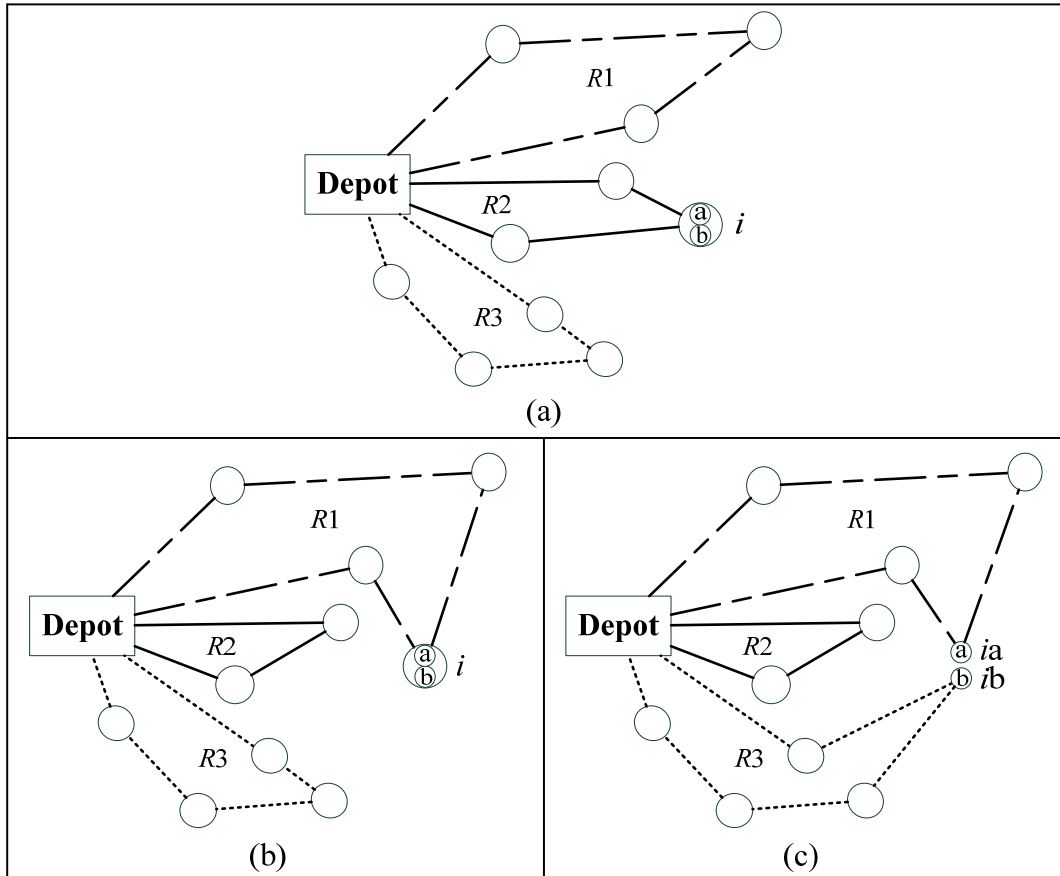


圖 7  $(1-0)^*$  顧客物品分開運送示意圖

### 3.3 GLNPSO 求解程序

粒子  $i$  在第  $t$  個迭代時經過編解碼模組得到粒子初始解  $S_i(t)$ ，隨即透過鄰域改善模組優化為解  $S_i^*(t)$ ，並利用 MCVRP 的目標函數 (式 6) 計算解  $S_i^*(t)$  的成本值作為粒子  $i$  在第  $t$  個迭代的適應值  $\varphi(X_i(t))$ 。當粒子群中的所有粒子皆完成計算適應值後，便開始對各個粒子進行 GLNPSO 學習，更新粒子的位置。求解的主要步驟敘述如下：

- 步驟 1. (初始化參數)：設定迭代數  $t = 1$ ，總迭代次數  $T$ ，粒子群大小  $I$ 。
- 步驟 2. (初始化粒子群)：產生  $I$  個粒子編碼  $X_i$  作為粒子群， $i = 1 \dots I$ ；分別設定各粒子  $i$  各維度  $d$  的起始速度  $v_{id}(1) = 0$ ，及其個別最佳解與群體最佳解的適應值分別為  $\varphi(P_i) = \infty$  與  $\varphi(P_g) = \infty$ 。
- 步驟 3. (解碼產生路線)：對各粒子  $i$  的編碼  $X_i(t)$ ，進行解碼以產生其路線解  $S_i(t)$ 。
- 步驟 4. (VND 鄰域改善)：對各  $S_i(t)$  以 VND 模組進行改善，得到其局域最佳解  $S_i^*(t)$  並計算其路線成本作為該粒子的適應值  $\varphi(X_i(t))$ 。
- 步驟 5. (更新 pbest  $P_i$ )：若  $\varphi(X_i(t)) < \varphi(P_i)$ ，則更新粒子  $i$  個別最佳解，即  $P_i = X_i(t)$ 。
- 步驟 6. (更新 gbest  $P_g$ )：若  $\varphi(P_i) < \varphi(P_g)$ ，則更新所有粒子的群體最佳解，即  $P_g = P_i$ 。
- 步驟 7. (更新 lbest  $P_i^L$ )：對各粒子  $i$ ，挑選  $\varphi(P_j)$  最接近  $\varphi(X_i(t))$  的  $K$  個粒子  $j$ ， $j \neq i$ ，設定其中  $\varphi(P_j)$  最小的  $P_j$  作為粒子  $i$  的  $P_i^L$ 。
- 步驟 8. (更新 nbest  $P_{id}^N$ )：對各粒子  $i$  的各個維度  $d$ ，計算其他粒子相對應的  $FDR$  值，如式 (3)，選取其最大值的粒子  $\alpha$  為粒子  $i$  的鄰近最佳解 (nbest)，並更新  $P_{id}^N = P_{\alpha d}$ 。
- 步驟 9. (粒子位移更新)：利用式 (5) 計算各個粒子  $i$  的各維度  $d$  的位移速度  $v_{id}(t+1)$ ，再以式 (2) 更新其位置  $x_{id}(t+1)$ 。在式 (5) 中的  $w(t)$  權重值部分本研究採用式 (3) 隨迭代數增加而遞減。
- 步驟 10. (停止條件)：若  $t = T$ ，則停止，輸出  $\varphi(P_g)$ ；否則， $t = t+1$  並返回步驟 3。

## 四、MCVRP 標竿題庫測試結果比較分析

### 4.1 MCVRP 標竿題庫說明

目前文獻上有兩套 MCVRP 的標竿測試題庫，以下稱為題庫 1 (Set 1) 與題庫 2 (Set 2)，均由 Fallahi 等人<sup>[18]</sup>提出。兩套題庫均考慮兩種物品 ( $p = 2$ )，各 20 題測試題，顧客點數範圍從 50 點至 483 點。這 20 題例題皆是依據 Christofides 等人<sup>[12]</sup>的 14 題以及 Golden 等人<sup>[32]</sup>的 6 題傳統 VRP 的測試例題修改而成。兩組題庫總共 40 個測試例題的相關資訊如表 3 所示，包含例題名稱 (instances)、顧客數 ( $n$ )、服務時間長度限制 ( $L$ ) 以及兩種物品對應的貨艙容量資訊， $Q_1$  與  $Q_2$ ，的各項資訊。

題庫 1 與題庫 2 兩組題庫的各個例題，均保持原 VRP 測試例題的顧客數 ( $n$ )，各顧客的位置與總需求量 ( $q_i$ )，以及車輛總容量 ( $Q$ ) 與車輛服務的時間限制 ( $L$ )。兩題庫的差異主要在於各顧客對兩種物品的需求， $q_{i1}$  與  $q_{i2}$ ，以及各車輛乘載兩種物品的貨艙容量，即  $Q_1$  與  $Q_2$  的不同設計。具體而言，題庫 1 的設計是把  $Q$  與  $q_i$  平分給兩種物品，即  $Q_1 = Q_2 = Q/2$ ，且  $q_{i1} = q_{i2} = q_i/2$ 。題庫 2 則是以隨機方式把  $q_i$  分給  $q_{i1}$  與  $q_{i2}$ ，即選取  $[3, 5]$  區間內的隨機整數  $k$ ，先設定  $q_{i1} = q_i/k$  與  $q_{i2} = q_i - q_{i1}$ 。再根據所有顧客  $q_{i1}$  與  $q_{i2}$  的平均值，即  $\overline{Q_1}$  與  $\overline{Q_2}$  值，設定  $Q_1 = (Q \times \overline{Q_1}) / (\overline{Q_1} + \overline{Q_2})$  與  $Q_2 = (Q \times \overline{Q_2}) / (\overline{Q_1} + \overline{Q_2})$ ，故  $q_{i1} \neq q_{i2}$  且  $\overline{Q_1} \neq \overline{Q_2}$ 。簡而言之，Fallahi 等人<sup>[18]</sup>以「均分」以及「非均分」兩種不同的情境分別產生題庫 1 與題庫 2。

表 3 MCVRP 標竿題庫資訊

| Instances | $n$ | $L$  | Set 1 |       | Set 2   |          |
|-----------|-----|------|-------|-------|---------|----------|
|           |     |      | $Q_1$ | $Q_2$ | $Q_1$   | $Q_2$    |
| Vrpnc1    | 50  | —    | 80    | 80    | 56.99   | 103.01   |
| Vrpnc2    | 75  | —    | 70    | 70    | 50.11   | 89.89    |
| Vrpnc3    | 100 | —    | 100   | 100   | 70.99   | 129.01   |
| Vrpnc4    | 150 | —    | 100   | 100   | 71.71   | 128.29   |
| Vrpnc5    | 199 | —    | 100   | 100   | 71.68   | 128.32   |
| Vrpnc6    | 50  | 200  | 80    | 80    | 56.99   | 103.01   |
| Vrpnc7    | 75  | 160  | 70    | 70    | 50.11   | 89.89    |
| Vrpnc8    | 100 | 230  | 100   | 100   | 70.99   | 129.01   |
| Vrpnc9    | 150 | 200  | 100   | 100   | 71.71   | 128.29   |
| Vrpnc10   | 199 | 200  | 100   | 100   | 71.68   | 128.32   |
| Vrpnc11   | 120 | —    | 100   | 100   | 72.00   | 128.00   |
| Vrpnc12   | 100 | —    | 100   | 100   | 70.70   | 129.30   |
| Vrpnc13   | 120 | 720  | 100   | 100   | 72.00   | 128.00   |
| Vrpnc14   | 100 | 1040 | 100   | 100   | 70.70   | 129.30   |
| E072-04f  | 71  | —    | 15000 | 15000 | 9668.56 | 20331.40 |
| E076-07u  | 75  | —    | 110   | 110   | 78.74   | 141.26   |
| E076-08s  | 75  | —    | 90    | 90    | 64.43   | 115.57   |
| E135-07f  | 134 | —    | 1105  | 1105  | 689.89  | 1520.11  |
| E241-22k  | 240 | —    | 100   | 100   | 71.03   | 128.97   |
| E484-19k  | 483 | —    | 500   | 500   | 350.97  | 649.03   |

## 4.2 PSO 參數與 VND 執行次序之設定

PSO 演算法執行時牽涉到的參數，包括 GLNPSO 粒子學習機制相關的參數以及演化過程需要定義的迭代數以及粒子數。首先，GLNPSO 學習相關的參數包含初始慣性  $w(1)$ 、最終慣性  $w(T)$ 、鄰近粒子數  $K$  以及學習係數  $c_p$ 、 $c_g$ 、 $c_l$  與  $c_n$ 。本研究依據 Ai 與 Kachitvichyanukul<sup>[5, 11]</sup>、Kachitvichyanukul 等人<sup>[7]</sup> 以及韓復華與楊禮瑛<sup>[6]</sup> 等 4 篇文獻應用 GLNPSO 機制的成功經驗，分別設定  $w(1) = 0.9$ 、 $w(T) = 0.4$ 、 $K = 5$ 、 $c_p = 0.5$ 、 $c_g = 0.5$ 、 $c_l = 1.5$  與  $c_n = 1.5$ ，並以這些參數設定進行有關演化需要的迭代數與粒子數參數測試，說明如下。

在 PSO 演算法中，迭代數與粒子數的設定對於求解的表現上有一定程度的影響。一般而言，迭代數與粒子數對應求解結果的精確度以及運算時間呈現正比關係，意即當迭代數與粒子數設定值越高時，結果的精確度越佳但求解的時間也越長。在 Ai 與 Kachitvichyanukul<sup>[5, 11]</sup> 的兩篇研究中建議迭代數為 1,000，粒子數為 50。韓復華與楊禮瑛<sup>[6]</sup> 則建議迭代數為 1,000，粒子數為 20。故本研究綜合參考其建議，對迭代數 ( $T$ ) 與粒子數 ( $I$ ) 兩個參數，分別在  $T = 500$ 、 $750$  與  $1,000$  以及  $I = 10$ 、 $20$ 、 $30$  與  $50$  的範圍進行測試，在同時考慮求解績效與運算時間的表現後，以迭代數  $T = 750$  以及粒子數  $I = 30$  作為執行 PSO 演算法使用之參數設定。本研究 PSO 演算法所有相關參數的設定值彙整如表 4 所示。

表 4 PSO 演算法相關參數設定

| 參數         | 設定值        |
|------------|------------|
| 粒子數        | $I=30$     |
| 迭代數        | $T=750$    |
| 鄰近粒子數      | $K=5$      |
| 初始慣性值      | $w(1)=0.9$ |
| 最終慣性值      | $w(T)=0.4$ |
| pbest 學習係數 | $c_p=0.5$  |
| gbest 學習係數 | $C_g=0.5$  |
| lbest 學習係數 | $C_l=1.5$  |
| nbest 學習係數 | $C_n=1.5$  |

除了 PSO 演算法相關的參數，VND 改善模組內的各種交換法執行次序也必須予以設定。如 3.2 節所述，本研究採用的 VND 共有兩種路線內與 6 種路線間的交換改善法，其中 5 種路線間的交換法，即  $(1, 0)$ 、 $(1, 1)$ 、 $(2, 0)$ 、 $(2, 1)$  與  $(2, 2)$  交換法，可撰寫成 1 個  $\lambda = 2$  的  $\lambda$ -interchanges 次模組。如此，VND 模組可簡化為 2-opt、Or-opt、2-opt\* 與  $\lambda$ -interchanges 4 個次模組，執行上共有  $4! = 24$  種不同的組合。本研究對這 24 種組合進行測試，結果發現 (Or-opt、2-opt\*、 $\lambda$ -interchanges、2-opt) 的執行組合平均績效最好，故將其設定為 PSO

執行時採用的 VND 次序。有關 PSO 演算法與 VND 執行次序測試的細節，可參閱林致瑄的碩士論文<sup>[33]</sup>。

### 4.3 測試結果與績效分析

本研究 PSO 演算法的電腦程式以 C# 語言撰寫，在 Win 7 作業系統下，使用 i7-3770 (3.40GHz) CPU 的個人電腦執行各標準題庫之測試。PSO 測試執行的樣本數為 30 次，以其最佳結果代表 PSO 之績效。所有 PSO 對各題庫不同條件下的測試結果彙整於表 5 至表 8。各表共同的結構如下：前 3 欄分別是各測試題 (instance) 的名稱，顧客點數 ( $n$ ) 與其目前文獻最佳解 (best known solution, BKS) 的成本。最後 3 欄代表本研究 PSO 演算法對各題的測試結果，包括其成本 (Cost)、與 BKS 之間的百分比誤差 (Gap) 以及電腦運算之 CPU 時間 (Time)。各表中的粗黑字體代表各測試題更新包含 PSO 結果後的現行最佳解 (current best known solution, CBKS) 對應之成本。其餘中間的欄位則是與 PSO 進行績效比較的其他 MCVRP 求解演算法的求解績效 (Cost、Gap 與 Time)。由於各演算法在程式撰寫以及測試時，所採用的硬體設備皆有不同，因此在其他演算法的 Time 欄位所提供的數值是直接引用於對應之文獻數據，並以附註說明各演算法進行實驗時所使用的 CPU 規格，以提供參考。

#### 4.3.1 題庫 1 測試結果與分析

題庫 1 在「不可分送」與「可分送」兩種條件下的測試結果，分別如表 5 與表 6 所示。如 4.1 節所述，題庫 1 各測試題是由若干傳統 VRP 測試題，將其顧客需求 ( $q_i$ ) 與車容量 ( $Q$ ) 平分為兩種物品設計而成。在此相等平分狀態下，任一物品的 VRP 路線也完全適用於另一種物品的配送。因此，對於題庫 1 的每個測試題而言，無論是「不可分送」或「可分送」的條件下，其對應的 VRP 解亦是該題 MCVRP 的可行解。同理，題庫 1 各題 MCVRP 文獻最佳解的來源必須考慮到其對應的 VRP 之 BKS 最佳解。事實上，如表 5 與表 6 所示，無論是「不可分送」或「可分送」的條件，VRP 目前的 BKS 也是 MCVRP 題庫 1 各測試題的 BKS 及 CBKS。

題庫 1 在「不可分送」配送條件下的測試結果如表 5 所示。現有文獻中可提供作為績效比較的對象不多，僅有 Fallahi 等人<sup>[18]</sup>的 MA 演算法與 TS 演算法以及 Mendoza 等人<sup>[25]</sup>求解隨機需求的 MCVRP 所提出的 MA\_sd 演算法。在此 20 例題的求解的績效上，結果顯示本研究之 PSO 方法在精確度方面明顯優於其他方法。具體而言，在 20 個測試題中，PSO 可求得 7 題目前文獻最佳解，平均誤差僅為 1.10%。其餘方法中以 TS 的求解結果最佳，可求得 2 題文獻最佳解，平均求解誤差為 2.94%。在平均求解時間上，以 MA\_sd (1.42 分鐘) 最佳，MA (2.53 分鐘) 次佳，本研究 PSO (16.83 分鐘) 相對耗時較長。

表 5 題庫 1\_「不可分送」求解結果與績效比較

| Instance   | n   | BKS cost                  | MA <sup>a</sup> |       |                          | TS <sup>a</sup> |       |                          | MA_sd <sup>b</sup> |       |                          | PSO           |      |                          |
|------------|-----|---------------------------|-----------------|-------|--------------------------|-----------------|-------|--------------------------|--------------------|-------|--------------------------|---------------|------|--------------------------|
|            |     |                           | Cost            | Gap   | Time <sup>c</sup> (min.) | Cost            | Gap   | Time <sup>c</sup> (min.) | Cost               | Gap   | Time <sup>d</sup> (min.) | Cost          | Gap  | Time <sup>e</sup> (min.) |
| Vrpnc1     | 50  | <b>524.6<sup>f</sup></b>  | <b>524.6</b>    | 0.00  | 0.29                     | <b>524.6</b>    | 0.00  | 0.33                     | <b>524.6</b>       | 0.00  | 0.16                     | <b>524.6</b>  | 0.00 | 0.72                     |
| Vrpnc2     | 75  | <b>835.3<sup>f</sup></b>  | 855.8           | 2.45  | 0.42                     | 850             | 1.76  | 0.94                     | 857.3              | 2.63  | 0.27                     | 836.4         | 0.13 | 2.10                     |
| Vrpnc3     | 100 | <b>826.1<sup>f</sup></b>  | 876.8           | 6.14  | 0.36                     | 831.3           | 0.63  | 0.85                     | 841.6              | 1.88  | 0.39                     | <b>826.1</b>  | 0.00 | 4.04                     |
| Vrpnc4     | 150 | <b>1028.4<sup>f</sup></b> | 1089.6          | 5.95  | 1.57                     | 1061.1          | 3.18  | 4.76                     | 1045.9             | 1.70  | 0.78                     | 1043.8        | 1.50 | 11.13                    |
| Vrpnc5     | 199 | <b>1291.4<sup>f</sup></b> | 1389.6          | 7.60  | 1.93                     | 1348.3          | 4.41  | 6.73                     | 1381.9             | 7.01  | 1.43                     | 1329.1        | 2.92 | 21.38                    |
| Vrpnc6     | 50  | <b>555.4<sup>f</sup></b>  | 571.4           | 2.88  | 1.02                     | 575.9           | 3.69  | 0.21                     | 556.7              | 0.23  | 0.14                     | <b>555.4</b>  | 0.00 | 0.82                     |
| Vrpnc7     | 75  | <b>909.7<sup>f</sup></b>  | 933.0           | 2.56  | 0.65                     | 970.8           | 6.72  | 0.36                     | 924.3              | 1.60  | 0.27                     | 915.0         | 0.58 | 2.60                     |
| Vrpnc8     | 100 | <b>865.9<sup>f</sup></b>  | 969.2           | 11.93 | 0.31                     | 888.6           | 2.62  | 1.01                     | 877.6              | 1.35  | 0.40                     | <b>865.9</b>  | 0.00 | 4.46                     |
| Vrpnc9     | 150 | <b>1162.5<sup>f</sup></b> | 1230.9          | 5.88  | 1.65                     | 1232.1          | 5.99  | 1.64                     | 1223.1             | 5.21  | 0.79                     | 1177.3        | 1.27 | 13.07                    |
| Vrpnc10    | 199 | <b>1395.8<sup>f</sup></b> | 1520.1          | 8.91  | 2.34                     | 1538.6          | 10.23 | 0.59                     | 1497.5             | 7.29  | 1.12                     | 1436.5        | 2.92 | 24.31                    |
| Vrpnc11    | 120 | <b>1042.1<sup>f</sup></b> | 1046.1          | 0.38  | 0.80                     | 1043.3          | 0.12  | 2.38                     | 1046.9             | 0.46  | 0.64                     | <b>1042.1</b> | 0.00 | 2.17                     |
| Vrpnc12    | 100 | <b>819.6<sup>f</sup></b>  | 820.6           | 0.12  | 0.30                     | <b>819.6</b>    | 0.00  | 1.05                     | 821.0              | 0.17  | 0.41                     | <b>819.6</b>  | 0.00 | 3.69                     |
| Vrpnc13    | 120 | <b>1541.1<sup>f</sup></b> | 1547.4          | 0.41  | 1.27                     | 1582.2          | 2.67  | 0.74                     | 1586.9             | 2.97  | 0.74                     | 1547.7        | 0.43 | 7.45                     |
| Vrpnc14    | 100 | <b>866.4<sup>f</sup></b>  | 866.9           | 0.06  | 0.39                     | 868.6           | 0.25  | 0.73                     | 868.1              | 0.20  | 0.40                     | <b>866.4</b>  | 0.00 | 3.64                     |
| E072-04f   | 71  | <b>237.0<sup>f</sup></b>  | 241.9           | 2.07  | 0.20                     | 244.5           | 3.16  | 0.45                     | 242.0              | 2.11  | 0.32                     | 241.9         | 2.07 | 1.68                     |
| E076-07u   | 75  | <b>682.5<sup>f</sup></b>  | 703.5           | 3.08  | 0.25                     | 692.2           | 1.42  | 0.42                     | 698.8              | 2.39  | 0.26                     | 687.5         | 0.73 | 2.11                     |
| E076-08s   | 75  | <b>735.0<sup>f</sup></b>  | 761.9           | 3.66  | 0.26                     | 748.5           | 1.84  | 0.41                     | 757.1              | 3.01  | 0.27                     | 740.6         | 0.76 | 2.17                     |
| E135-07f   | 134 | <b>1162.9<sup>f</sup></b> | 1193.2          | 2.61  | 0.79                     | 1174.4          | 0.99  | 1.60                     | 1195.3             | 2.79  | 0.98                     | 1163.3        | 0.03 | 8.98                     |
| E241-22k   | 240 | <b>707.7<sup>g</sup></b>  | 748.6           | 5.78  | 8.41                     | 741.4           | 4.76  | 6.45                     | 763.5              | 7.88  | 1.83                     | 723.4         | 2.22 | 33.58                    |
| E484-19k   | 483 | <b>1101.5<sup>g</sup></b> | 1250.1          | 13.49 | 27.39                    | 1149.5          | 4.36  | 59.86                    | 1250.9             | 13.56 | 16.84                    | 1171.3        | 6.34 | 186.69                   |
| Average    |     | 914.5                     | 957.1           | 4.30  | 2.53                     | 944.3           | 2.94  | 4.57                     | 948.1              | 3.22  | 1.42                     | 925.7         | 1.10 | 16.83                    |
| CBKS count |     | 20                        | 1               |       |                          | 2               |       |                          | 1                  |       |                          | 7             |      |                          |

<sup>a</sup> Fallahi 等人<sup>[18]</sup>；<sup>b</sup> Mendoza 等人<sup>[25]</sup>；<sup>c</sup> P4, 2.4 GHz；<sup>d</sup> 2.4 GHz；<sup>e</sup> i7-3770, 3.4 GHz；<sup>f</sup> VRP Web；<sup>g</sup> Jianyong 等人<sup>[34]</sup>。

表 6 呈現的是題庫 1 在「可分送」條件下的測試結果。在此可作為績效比較的對象除了前述的 MA 與 TS 演算法以外，亦包括 Muyldermans 等人<sup>[24]</sup> 的 GLS 演算法。以完整 20 題的測試結果而言，PSO 在求解精確度方面表現最佳，可求得 7 題文獻最佳解，平均誤差僅為 1.10%。TS 與 MA 則在伯仲之間，TS 的平均誤差 (2.02%) 優於 MA (2.32%)，但 MA (2 題) 較 TS (1 題) 多求得 1 題 BKS。由於 GLS 未考慮路線時間長度的限制條件，且只能處理整數需求的測試題，故其僅在此題庫中測試了 9 個題目。以 GLS 求解之 9 題例題進行比較，PSO 之平均誤差為 1.41%，稍微劣於 GLS 演算法的 1.38%。求解時間方面，以 20 題平均而言，MA (3.78 分鐘) 最佳，TS (8.43 分鐘) 次之，PSO 時間最長 (38.88 分鐘)。

對各題庫 1 的測試題而言，因其是由 VRP 對稱平分設計產生，每個 MCVRP 實際上都是由兩個相同而獨立的 VRP 子問題合併而成。在此情況下，任一物品 VRP 子問題的最佳解其實也是另一物品 VRP 子問題的最佳解。此時，MCVRP 在「不可分送」與「可分送」兩種配送條件下的最佳結果並無差異。換言之，兩種配送條件下的最佳解結果都是合併而不分送的路線型態，因兩個物品 VRP 的最佳路線是完全相同，如此就不會產生把同一顧客的兩種物品分送到兩條路線上的「分送」結果。本研究以 PSO 求解題庫 1 在「可分送」

條件下的測試結果 (見表 6) 的路線圖經過檢視，亦發現其 20 題的結果都合理地具有「不可分送」的路線型態。

表 6 題庫 1\_「可分送」求解結果與績效比較

| Instance               | n   | BKS cost                  | MA <sup>a</sup> |      |                          | TS <sup>a</sup> |      |                          | GLS <sup>b</sup> |      |                          | PSO           |      |                          |
|------------------------|-----|---------------------------|-----------------|------|--------------------------|-----------------|------|--------------------------|------------------|------|--------------------------|---------------|------|--------------------------|
|                        |     |                           | Cost            | Gap  | Time <sup>c</sup> (min.) | Cost            | Gap  | Time <sup>c</sup> (min.) | Cost             | Gap  | Time <sup>d</sup> (min.) | Cost          | Gap  | Time <sup>e</sup> (min.) |
| Vrpnc1                 | 50  | <b>524.6<sup>f</sup></b>  | <b>524.6</b>    | 0.00 | 0.40                     | <b>524.6</b>    | 0.00 | 0.26                     | <b>524.6</b>     | 0.00 | 8.97                     | <b>524.6</b>  | 0.00 | 1.50                     |
| Vrpnc2                 | 75  | <b>835.3<sup>f</sup></b>  | 842.7           | 0.89 | 0.92                     | 851.8           | 1.98 | 0.36                     | 837.4            | 0.25 | 8.61                     | 836.4         | 0.13 | 4.31                     |
| Vrpnc3                 | 100 | <b>826.1<sup>f</sup></b>  | 853.2           | 3.28 | 1.04                     | 835.2           | 1.10 | 1.59                     | 829.8            | 0.45 | 9.41                     | <b>826.1</b>  | 0.00 | 7.85                     |
| Vrpnc4                 | 150 | <b>1028.4<sup>f</sup></b> | 1070.9          | 4.13 | 3.45                     | 1055.1          | 2.60 | 6.85                     | 1040.2           | 1.15 | 10.11                    | 1040.4        | 1.17 | 21.83                    |
| Vrpnc5                 | 199 | <b>1291.4<sup>f</sup></b> | 1330.3          | 3.01 | 6.84                     | 1348.8          | 4.44 | 13.49                    | 1314.0           | 1.75 | 10.2                     | 1326.4        | 2.71 | 47.53                    |
| Vrpnc6                 | 50  | <b>555.4<sup>f</sup></b>  | 559.3           | 0.70 | 0.32                     | 560.0           | 0.83 | 0.24                     |                  |      |                          | <b>555.4</b>  | 0.00 | 1.77                     |
| Vrpnc7                 | 75  | <b>909.7<sup>f</sup></b>  | 928.1           | 2.02 | 0.77                     | 937.1           | 3.01 | 0.90                     |                  |      |                          | 912.9         | 0.35 | 5.16                     |
| Vrpnc8                 | 100 | <b>865.9<sup>f</sup></b>  | 892.2           | 3.04 | 0.95                     | 894.5           | 3.30 | 1.75                     |                  |      |                          | <b>865.9</b>  | 0.00 | 8.70                     |
| Vrpnc9                 | 150 | <b>1162.5<sup>f</sup></b> | 1199.9          | 3.22 | 2.62                     | 1184.2          | 1.87 | 6.59                     |                  |      |                          | 1180.3        | 1.53 | 25.92                    |
| Vrpnc10                | 199 | <b>1395.8<sup>f</sup></b> | 1461.2          | 4.69 | 8.35                     | 1467.6          | 5.14 | 6.59                     |                  |      |                          | 1443.5        | 3.42 | 50.29                    |
| Vrpnc11                | 120 | <b>1042.1<sup>f</sup></b> | 1044.7          | 0.25 | 2.11                     | 1043.8          | 0.16 | 1.08                     | 1048.7           | 0.63 | 8.33                     | <b>1042.1</b> | 0.00 | 10.99                    |
| Vrpnc12                | 100 | <b>819.6<sup>f</sup></b>  | <b>819.6</b>    | 0.00 | 1.28                     | 822.0           | 0.29 | 0.58                     | <b>819.6</b>     | 0.00 | 9.31                     | <b>819.6</b>  | 0.00 | 6.10                     |
| Vrpnc13                | 120 | <b>1541.1<sup>f</sup></b> | 1541.2          | 0.01 | 2.33                     | 1542.9          | 0.12 | 2.80                     |                  |      |                          | 1548.7        | 0.49 | 13.97                    |
| Vrpnc14                | 100 | <b>866.4<sup>f</sup></b>  | 866.8           | 0.05 | 0.93                     | 867.1           | 0.08 | 0.70                     |                  |      |                          | <b>866.4</b>  | 0.00 | 6.91                     |
| E072-04f               | 71  | <b>237.0<sup>f</sup></b>  | 244.8           | 3.29 | 0.19                     | 241.9           | 2.07 | 0.47                     |                  |      |                          | 241.9         | 2.07 | 2.99                     |
| E076-07u               | 75  | <b>682.5<sup>f</sup></b>  | 690.8           | 1.22 | 0.47                     | 690.2           | 1.13 | 0.49                     |                  |      |                          | 687.6         | 0.75 | 3.92                     |
| E076-08s               | 75  | <b>735.0<sup>f</sup></b>  | 749.5           | 1.97 | 0.54                     | 747.9           | 1.76 | 0.38                     |                  |      |                          | 740.7         | 0.78 | 4.09                     |
| E135-07f               | 134 | <b>1162.9<sup>f</sup></b> | 1165.6          | 0.23 | 2.67                     | 1179.2          | 1.40 | 2.35                     |                  |      |                          | 1163.3        | 0.03 | 17.06                    |
| E241-22k               | 240 | <b>707.7<sup>g</sup></b>  | 743.8           | 5.10 | 5.11                     | 741.2           | 4.73 | 13.43                    | 719.7            | 1.70 | 10.97                    | 723.2         | 2.19 | 72.76                    |
| E484-19k               | 483 | <b>1101.5<sup>g</sup></b> | 1204.1          | 9.31 | 34.27                    | 1148.9          | 4.30 | 107.66                   | 1173.4           | 6.53 | 18.13                    | 1171.9        | 6.39 | 463.86                   |
| Average                |     | 914.5                     | 936.7           | 2.32 | 3.78                     | 934.2           | 2.02 | 8.43                     |                  |      |                          | 925.9         | 1.10 | 38.88                    |
| Average_9 <sup>h</sup> |     | 908.5                     | 937.1           | 2.89 | 6.16                     | 930.2           | 2.18 | 16.14                    | 923.0            | 1.38 | 10.45                    | 934.4         | 1.40 | 70.75                    |
| CBKS count             |     | 20                        | 2               |      |                          | 1               |      |                          | 2                |      |                          | 7             |      |                          |

<sup>a</sup> Fallahi 等人<sup>[18]</sup>；<sup>b</sup> Muyldermans 等人<sup>[24]</sup>；<sup>c</sup> P4, 2.4 GHz；<sup>d</sup> Pentium M740, 1.73 GHz；<sup>e</sup> i7-3770, 3.40 GHz；

<sup>f</sup> VRP Web；<sup>g</sup> Jianyong 等人<sup>[34]</sup>；<sup>h</sup> the average of the 9 instances tested by GLS。

#### 4.3.2 題庫 2 測試結果與分析

題庫 2 是以隨機分配設計產生兩種物品需求的測試題。由於不具備對稱性，其「不可分送」與「可分送」兩種條件下的 (最佳) 結果，除非是特例，否則應有不同。首先說明本研究 PSO 測試 MCVRP 題庫 2 在「不可分送」條件下的相關結果。該情境下，與 PSO 做績效比較的對象包括前小節所述的 MA，TS 與 MA\_sd 等 3 種方法；各測試題的 BKS 即為 3 種方法結果的最佳者。如表 7 所示，在 20 個測試題中，目前文獻的 BKS 有 3 題，10 題與 7 題，分別由 MA，TS 與 MA\_sd 所締造的。然而，經過與本研究 PSO 的結果比較後，發現 20 題中有 17 題的 BKS 被 PSO 突破，加上與現有 BKS 平手的 1 題，PSO 共擁有 18 題最新的 CBKS 紀錄。PSO 測試精確度方面的優越亦表現在平均誤差上，其與文獻 BKS 最佳成本的平均誤差為 -1.09%。其他 3 個演算法的精確度均不如 PSO，其中 MA\_sd 的表

現尚佳，平均誤差為 1.75%，MA 與 TS 兩者的平均誤差則均為 2.19%。就求解時間而言，MA\_sd 執行時間最短，平均每題 1.25 分鐘。MA 與 TS 次之，平均每題分別是 2.52 與 2.60 分鐘。PSO 時間最長，每題平均要 17 分鐘。

表 7 題庫 2\_「不可分送」求解結果與績效比較

| Instance   | n   | BKS cost      | MA <sup>a</sup> |      |                          | TS <sup>a</sup> |       |                          | MA_sd <sup>b</sup> |      |                          | PSO           |       |                          |
|------------|-----|---------------|-----------------|------|--------------------------|-----------------|-------|--------------------------|--------------------|------|--------------------------|---------------|-------|--------------------------|
|            |     |               | Cost            | Gap  | Time <sup>c</sup> (min.) | Cost            | Gap   | Time <sup>c</sup> (min.) | Cost               | Gap  | Time <sup>d</sup> (min.) | Cost          | Gap   | Time <sup>e</sup> (min.) |
| Vrpnc1     | 50  | 550.2         | 558.8           | 1.56 | 0.29                     | 556.1           | 1.07  | 0.26                     | 550.2              | 0.00 | 0.16                     | <b>549.5</b>  | -0.13 | 0.78                     |
| Vrpnc2     | 75  | 863.6         | 888.6           | 2.89 | 0.43                     | 863.6           | 0.00  | 0.23                     | 884.3              | 2.40 | 0.29                     | <b>863.0</b>  | -0.07 | 2.17                     |
| Vrpnc3     | 100 | 837.6         | 878.4           | 4.87 | 0.36                     | 837.6           | 0.00  | 0.66                     | 852.7              | 1.80 | 0.42                     | <b>830.2</b>  | -0.89 | 4.06                     |
| Vrpnc4     | 150 | 1070.7        | 1089.1          | 1.72 | 1.57                     | 1070.7          | 0.00  | 1.83                     | 1080.0             | 0.87 | 0.84                     | <b>1058.4</b> | -1.15 | 11.16                    |
| Vrpnc5     | 199 | 1361.4        | 1408.5          | 3.46 | 1.93                     | 1361.4          | 0.00  | 3.47                     | 1403.5             | 3.09 | 1.52                     | <b>1359.0</b> | -0.18 | 21.31                    |
| Vrpnc6     | 50  | 562.7         | 569.4           | 1.19 | 0.28                     | 563.4           | 0.12  | 0.17                     | 562.7              | 0.00 | 0.14                     | <b>557.5</b>  | -0.92 | 0.87                     |
| Vrpnc7     | 75  | 946.3         | 955.1           | 0.93 | 0.65                     | 949.0           | 0.29  | 0.37                     | 946.3              | 0.00 | 0.27                     | <b>931.2</b>  | -1.60 | 2.49                     |
| Vrpnc8     | 100 | 874.7         | 958.9           | 9.63 | 0.31                     | 916.2           | 4.74  | 0.31                     | 874.7              | 0.00 | 0.40                     | <b>866.9</b>  | -0.89 | 4.34                     |
| Vrpnc9     | 150 | 1240.5        | 1262.7          | 1.79 | 1.65                     | 1290.8          | 4.05  | 0.14                     | 1240.5             | 0.00 | 0.86                     | <b>1181.1</b> | -4.79 | 12.51                    |
| Vrpnc10    | 199 | 1490.2        | 1509.1          | 1.27 | 2.34                     | 1490.2          | 0.00  | 3.17                     | 1515.6             | 1.70 | 1.30                     | <b>1451.1</b> | -2.62 | 23.63                    |
| Vrpnc11    | 120 | 1122.9        | 1122.9          | 0.00 | 0.80                     | 1201.6          | 7.01  | 0.47                     | 1162.4             | 3.52 | 0.80                     | <b>1118.5</b> | -0.39 | 5.62                     |
| Vrpnc12    | 100 | 916.6         | 926.5           | 1.08 | 0.30                     | 934.1           | 1.91  | 0.26                     | 916.6              | 0.00 | 0.39                     | <b>896.9</b>  | -2.15 | 3.66                     |
| Vrpnc13    | 120 | <b>1542.4</b> | <b>1542.4</b>   | 0.00 | 1.27                     | 1582.3          | 2.59  | 0.37                     | 1607.4             | 4.21 | 0.62                     | 1549.5        | 0.46  | 7.42                     |
| Vrpnc14    | 100 | 937.6         | 966.5           | 3.08 | 0.39                     | 1141.6          | 21.76 | 0.60                     | 937.6              | 0.00 | 0.39                     | <b>931.0</b>  | -0.70 | 3.68                     |
| E072-04f   | 71  | <b>262.3</b>  | 263.6           | 0.50 | 0.20                     | <b>262.3</b>    | 0.00  | 0.09                     | 262.5              | 0.08 | 0.37                     | <b>262.3</b>  | 0.00  | 2.21                     |
| E076-07u   | 75  | 697.8         | 702.2           | 0.63 | 0.25                     | 697.8           | 0.00  | 0.28                     | 704.1              | 0.90 | 0.26                     | <b>693.3</b>  | -0.64 | 2.10                     |
| E076-08s   | 75  | 772.2         | 793.5           | 2.76 | 0.26                     | 772.2           | 0.00  | 0.23                     | 778.2              | 0.78 | 0.25                     | <b>748.4</b>  | -3.08 | 2.12                     |
| E135-07f   | 134 | 1233.2        | 1233.2          | 0.00 | 1.24                     | 1235.2          | 0.16  | 0.87                     | 1293.8             | 4.91 | 0.81                     | <b>1219.8</b> | -1.09 | 9.24                     |
| E241-22k   | 240 | 787.8         | 796.7           | 1.13 | 8.41                     | 787.8           | 0.00  | 3.38                     | 810.7              | 2.91 | 1.94                     | <b>772.4</b>  | -1.95 | 34.07                    |
| E484-19k   | 483 | <b>1177.3</b> | 1240.9          | 5.40 | 27.39                    | <b>1177.3</b>   | 0.00  | 35.38                    | 1269.0             | 7.79 | 13.04                    | 1189.8        | 1.06  | 187.41                   |
| Average    |     | 964.2         | 983.4           | 2.19 | 2.52                     | 984.6           | 2.19  | 2.60                     | 982.6              | 1.75 | 1.25                     | 951.5         | -1.09 | 17.00                    |
| BKS count  |     | 20            | 3               |      |                          | 10              |       |                          | 7                  |      |                          | 1             |       |                          |
| CBKS count |     |               |                 |      |                          |                 |       |                          |                    |      |                          | 18            |       |                          |

<sup>a</sup> Fallahi 等人<sup>[18]</sup>；<sup>b</sup> Mendoza 等人<sup>[25]</sup>；<sup>c</sup> P4, 2.4 GHz；<sup>d</sup> 2.4 GHz；<sup>e</sup> i7-3770, 3.40 GHz。

題庫 2 在「可分送」條件下的測試結果如表 8，其中與 PSO 做績效比較的對象僅有 MA 與 TS 兩者。20 個測試題對目前文獻的 BKS 分別有 8 題與 12 題，由 MA 與 TS 兩個方法所締造。與「不可分送」條件下的優異求解表現相同，PSO 在 20 題中，突破了 17 題現有的 BKS，加上與現有 BKS 平手的 1 題，PSO 在「可分送」配送條件下，亦擁有 18 題最新的 CBKS 紀錄。PSO 與現有 BKS 成本的平均誤差為 -0.78%，對其他方法而言，TS 與 MA 的平均誤差分別為 0.38%與 0.78%。在求解時間方面，則是 MA 居首 (3.46 分鐘)，TS 次之 (6.13 分鐘)，PSO 墊後 (38.96 分鐘)。



表 8 題庫 2\_「可分送」求解結果與績效比較

| Instances  | <i>n</i> | BKS           | MA <sup>a</sup> |      |                             | TS <sup>a</sup> |      |                             | PSO           |       |                             |
|------------|----------|---------------|-----------------|------|-----------------------------|-----------------|------|-----------------------------|---------------|-------|-----------------------------|
|            |          |               | Cost            | Gap  | Time <sup>b</sup><br>(min.) | Cost            | Gap  | Time <sup>b</sup><br>(min.) | Cost          | Gap   | Time <sup>c</sup><br>(min.) |
| Vrpnc1     | 50       | 548.4         | 548.4           | 0.00 | 0.38                        | 550.5           | 0.38 | 0.21                        | <b>545.8</b>  | -0.47 | 1.50                        |
| Vrpnc2     | 75       | 873.6         | 874.5           | 0.10 | 0.75                        | 873.6           | 0.00 | 0.93                        | <b>857.7</b>  | -1.82 | 4.45                        |
| Vrpnc3     | 100      | 832.9         | 843.4           | 1.26 | 1.53                        | 832.9           | 0.00 | 2.65                        | <b>830.2</b>  | -0.32 | 7.91                        |
| Vrpnc4     | 150      | 1076.0        | 1078.1          | 0.20 | 2.65                        | 1076.0          | 0.00 | 5.78                        | <b>1057.4</b> | -1.73 | 21.72                       |
| Vrpnc5     | 199      | 1362.8        | 1368.8          | 0.44 | 8.12                        | 1362.8          | 0.00 | 6.51                        | <b>1354.9</b> | -0.58 | 46.83                       |
| Vrpnc6     | 50       | 558.6         | 558.7           | 0.02 | 0.45                        | 558.6           | 0.00 | 0.50                        | <b>557.5</b>  | -0.20 | 1.75                        |
| Vrpnc7     | 75       | 952.6         | 959.9           | 0.77 | 0.30                        | 952.6           | 0.00 | 0.56                        | <b>926.4</b>  | -2.75 | 5.16                        |
| Vrpnc8     | 100      | 890.1         | 890.1           | 0.00 | 1.69                        | 897.4           | 0.82 | 1.89                        | <b>866.9</b>  | -2.61 | 8.70                        |
| Vrpnc9     | 150      | 1186.2        | 1224.9          | 3.26 | 2.57                        | 1186.2          | 0.00 | 5.18                        | <b>1180.8</b> | -0.46 | 25.72                       |
| Vrpnc10    | 199      | 1475.8        | 1475.8          | 0.00 | 6.88                        | 1500.3          | 1.66 | 6.14                        | <b>1472.6</b> | -0.89 | 49.63                       |
| Vrpnc11    | 120      | 1113.4        | 1113.4          | 0.00 | 1.58                        | 1155.2          | 3.75 | 1.32                        | <b>1095.1</b> | -1.64 | 11.24                       |
| Vrpnc12    | 100      | 906.9         | 906.9           | 0.00 | 0.56                        | 907.3           | 0.04 | 1.52                        | <b>896.9</b>  | -1.10 | 7.00                        |
| Vrpnc13    | 120      | <b>1541.2</b> | <b>1541.2</b>   | 0.00 | 2.36                        | 1543.0          | 0.12 | 2.74                        | 1549.5        | 0.54  | 14.15                       |
| Vrpnc14    | 100      | 934.7         | 934.7           | 0.00 | 1.03                        | 941.4           | 0.72 | 0.89                        | <b>930.4</b>  | -0.46 | 7.73                        |
| E072-04f   | 71       | 262.5         | 262.5           | 0.00 | 0.21                        | 262.7           | 0.08 | 0.23                        | <b>262.3</b>  | -0.08 | 3.73                        |
| E076-07u   | 75       | 699.2         | 705.8           | 0.94 | 0.85                        | 699.2           | 0.00 | 0.36                        | <b>693.3</b>  | -0.84 | 4.01                        |
| E076-08s   | 75       | <b>748.4</b>  | 756.7           | 1.11 | 0.71                        | <b>748.4</b>    | 0.00 | 0.87                        | <b>748.4</b>  | 0.00  | 4.11                        |
| E135-07f   | 134      | 1248.3        | 1256.8          | 0.68 | 1.19                        | 1248.3          | 0.00 | 4.34                        | <b>1219.2</b> | -2.33 | 18.87                       |
| E241-22k   | 240      | 771.2         | 792.7           | 2.79 | 9.81                        | 771.2           | 0.00 | 15.32                       | <b>769.4</b>  | -0.23 | 76.52                       |
| E484-19k   | 483      | <b>1175.5</b> | 1224.0          | 4.13 | 25.77                       | <b>1175.5</b>   | 0.00 | 64.64                       | 1194.7        | 1.63  | 458.54                      |
| Average    |          | 957.9         | 965.9           | 0.78 | 3.46                        | 962.1           | 0.38 | 6.13                        | 950.0         | -0.78 | 38.96                       |
| BKS count  |          | 20            | 8               |      |                             | 12              |      |                             | 1             |       |                             |
| CBKS count |          |               |                 |      |                             |                 |      |                             | 18            |       |                             |

<sup>a</sup> Fallahi 等人<sup>[18]</sup>；<sup>b</sup> P4, 2.4 GHz；<sup>c</sup> i7-3770, 3.40 GHz。

#### 4.3.3 綜合分析與討論

綜觀 MCVRP 兩個題庫與兩種配送條件下的測試結果 (表 5 至表 8)，本研究提出的 PSO 演算法在 4 種情境 80 個測試題的平均誤差為 0.08%，優於 Fallahi 等人<sup>[18]</sup> 提出的 MA (2.40%) 與 TS (1.88%) 方法。同時，在這 80 個測試題中，PSO 求得 16 題文獻 BKS，並改善了 34 題；同時也在 80 題更新的 CBKS 中留下 50 題的紀錄。本研究將改善 BKS 幅度最大的兩題，題庫 2 Vrpnc9\_不可分送 (150 點) 與題庫 2 Vrpnc7\_可分送 (75 點) 的最佳結

果與路線圖呈現於附錄，完整的 50 題 MCVRP 最新的 CBKS 結果則放置於 <http://db.tt/FBxD94Gc>，供學術參考。

比較兩個題庫，PSO 在題庫 2 的績效明顯優於其在題庫 1 的表現；突破 BKS 的 34 題全部都是在題庫 2。這是由於題庫 1 的特殊設計，使得各測試題文獻最佳解皆是 VRP 的 BKS。而這些 VRP 的 BKS 是經過多年來諸多方法透過眾多參數校估後累積的結果，確實難以超越。本研究在題庫 1 的平均誤差為 1.01%，應該也可以算是不錯的績效表現。另一方面，VRP 畢竟是 MCVRP 的特例，目前 VRP BKS 的演算法無論再好，仍無法用來求解一般的 MCVRP。相對而言，題庫 2 還是測試 MCVRP 演算法比較客觀的一個標竿題庫。

MCVRP 具有「不可分送」與「可分送」的兩種問題型態，理論上而言，後者的限制條件較寬鬆，故其正確解 (exact solution) 的結果，一定是優於 (或等於) 前者。但由於本文探討與比較的方法，PSO、MA、TS、MA\_sd 與 GLS 都是啟發式解法，其求算的只是近似解 (approximate solution)，因此無法保證其具有「可分送」結果優於 (等於) 「不可分送」結果的性質。事實上，在兩題庫的測試皆有「不可分割」反而優於「可分割」結果的情況，如題庫 1：MA 的 E072-04f 題、TS 的 Vrpnc12 題以及 PSO 的 Vrpnc8 題 (見表 5 表 6)；題庫 2：MA 的 Vrpnc7 題、TS 的 Vrpnc2 題與 PSO 的 Vrpnc10 題 (見表 7 表 8)。雖然有此少數例外，各個演算法對大多數測試題還是能夠求得「可分送」比「不可分送」條件下較優 (或等於) 的結果。

我們進一步對「不可分送」與「可分送」的兩種問題型態進行數據分析。由於題庫 2 為測試 MCVRP 較為客觀的標竿題庫，因此我們於題庫 2 中挑選可分送優於不可分送最多的 Vrpnc11 以及可分送劣於不可分送最多的 Vrpnc10 兩個例題結果進行比較分析。發現 Vrpnc11 在可分送下 (平均) 乘載率可達 98.3%，確實較其不可分送時 (86.0%) 可以提高 12.3% 的乘載率；此外，Vrpnc11 可分送比不可分送的結果亦多節省一條路線。反之，對 Vrpnc10 的求解，可分送比不可分送的結果在乘載率與路線數兩方面均無改善。由此可觀察：乘載率是求解可分送條件 MCVRP 時的一個重要因素；亦因如此，本研究 PSO 演算法產生可分送結果劣於不可分送結果的原因，很可能是未將「乘載率」納入交換改善的考慮因素。

## 五、結論與建議

多貨艙車輛路線問題自 2008 年被提出後，因其實務應用價值，目前已逐漸受到重視。本研究首次採用 PSO 巨集啟發式方法求解該問題，並針對其「不可分送」與「可分送」兩種條件分別構建 MCSR-1 與 MCSR-2 兩套新的編解碼程序，並在 VND 改善模組中新設計一個 (1, 0)\*節點交換法以增加「可分送」求解時的深度搜尋，再以 GLNPSO 粒子學習策略執行演化做廣度之搜尋。以兩套國際標竿題庫測試的結果發現，本研究提出的 PSO 演算法極具競爭力。

本研究利用 GLNPSO 求解 MCVRP 的精確度績效甚佳，但其運算時間較長，尤其是在求解「可分送」問題時要花費近兩倍於求解「不可分送」問題的時間。這可能是 MCSR-2 的編碼總維度是 MCSR-1 的兩倍之故。此外，為了獲得良好的求解精確度以及穩定性，我們對每次迭代中的所有粒子解皆採用完整的 8 種交換法進行變動鄰域搜尋進行改善，此亦相對耗時。

本研究在各階段的演算法設計上，目前皆是以成本的節省作為基礎。經過觀察題庫 2 的 Vrpnc11 與 Vrpnc10 兩例題後，發現乘載率是求解可分送條件 MCVRP 時的一個重要因素。因此建議後續研究除了成本節省以外，亦應將乘載率納入可分送演算法考量。

綜觀上述，如何在保留求解的精確度之下，設計出更有效率且更適合問題特性的執行方式，即是為一個後續研究的方向。我們也期待未來有更多相關的研究，投入 MCVRP 這個重要的課題。

## 參考文獻

1. Kennedy, J. and Eberhart, R. C., "Particle Swarm Optimization", Proceedings of IEEE International Conference on Neural Networks, Vol. IV, Institute of Electrical and Electronics Engineers, 1995, pp. 1942-1948.
2. Shi, Y. and Eberhart, R. C., "A Modified Particle Swarm Optimizer", Evolutionary Computation Proceedings, IEEE World Congress on Computational Intelligence, 1998, pp. 69-73.
3. Eberhart, R. C. and Shi, Y., "Comparison between Genetic Algorithms And Particle Swarm Optimization", Proceedings of the 7th International Conference on Evolutionary Programming VII, INSTICC, 1998, pp. 611-616.
4. Pongchairerks, P. and Kachitvichyanukul, V., "A Non-homogenous Particle Swarm Optimization with Multiple Social Structures", Proceedings of International Conference on Simulation and Modeling, INSTICC, 2005, pp. A5-02.
5. Ai, T. J. and Kachitvichyanukul, V., "Particle Swarm Optimization and Two Solution Representations for Solving the Capacitated Vehicle Routing Problem", *Computers & Industrial Engineering*, Vol. 56, No. 1, 2009, pp. 380-387.
6. 韓復華、楊禮瑛，「應用粒子群演算法求解 OVRP 問題之研究」，*運輸學刊*，第 25 卷，第 2 期，民國 102 年，頁 199-220。
7. Kachitvichyanukul, V., Sombuntham, P., and Kunnappadeelert, S., "Two Solution Representations for Solving Multi-Depot Vehicle Routing Problem with Multiple Pickup and Delivery Requests via PSO", *Computers & Industrial Engineering*, Vol. 89, 2015, pp. 125-136.
8. Clerc, M., "The Swarm and the Queen: Towards a Deterministic and Adaptive Particle Swarm Optimization", Proceedings of the IEEE Congress on Evolutionary Computation, Vol. 3, Institute of Electrical and Electronics Engineers, 1999, pp. 1951-1957.

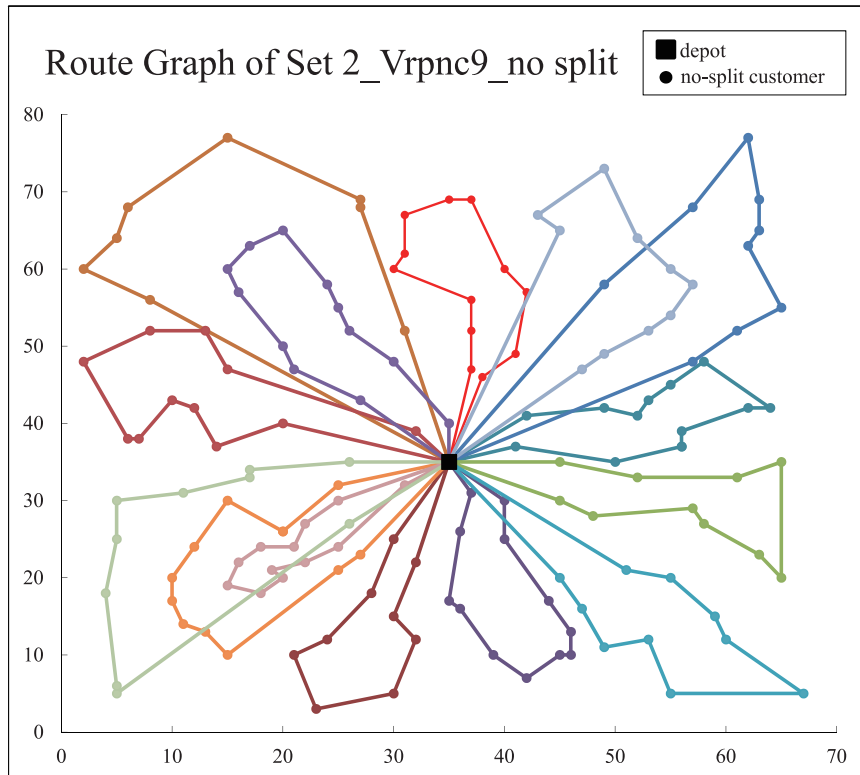
9. Clerc, M. and Kennedy, J., "The Particle Swarm—Explosion, Stability, and Convergence in a Multidimensional Complex Space", *IEEE Transactions on Evolutionary Computation*, Vol. 6, No. 1, 2002, pp. 58-73.
10. Chen, A. L., Yang, G. K., and Wu, Z. M., "Hybrid Discrete Particle Swarm Optimization Algorithm for Capacitated Vehicle Routing Problem", *Journal of Zhejiang University-Science A*, Vol. 7, No. 4, 2006, pp. 607-614.
11. Ai, T. J. and Kachitvichyanukul, V., "A Particle Swarm Optimization for the Vehicle Routing Problem with Simultaneous Pickup and Delivery", *Computers & Operations Research*, Vol. 36, No. 5, 2009, pp. 1693-1702.
12. Christofides, N., Mingozzi, A., and Toth, P., "The Vehicle Routing Problem", In: Christofides, N. et al. (Eds.), *Combinatorial Optimization*, Wiley, Chichester, UK, 1979, pp. 315-38.
13. Marinakis, Y., Marinaki M., and Dounias, G., "A Hybrid Particle Swarm Optimization Algorithm for the Vehicle Routing Problem", *Engineering Applications of Artificial Intelligence*, Vol. 23, No. 4, 2010, pp. 463-472.
14. Qi, C., "Application of Improved Discrete Particle Swarm Optimization in Logistics Distribution Routing Problem", *Procedia Engineering*, Vol. 15, 2011, pp. 3673-3677.
15. Goksal, F. P., Karaoglan, I., and Altiparmak, F., "A Hybrid Discrete Particle Swarm Optimization for Vehicle Routing Problem with Simultaneous Pickup and Delivery", *Computers & Industrial Engineering*, Vol. 65, No. 1, 2013, pp. 39-53.
16. Belmecheri, F., Prins, C., Yalaoui, F., and Amodeo, L., "Particle Swarm Optimization Algorithm for a Vehicle Routing Problem with Heterogeneous Fleet, Mixed Backhauls, and Time Windows", *Journal of Intelligent Manufacturing*, Vol. 24, No. 4, 2013, pp. 775-789.
17. Norouzi, N., Sadegh-Amalnick, M., Yalaoui, F., and Alinaghiyan, M., "Evaluating of the Particle Swarm Optimization in a Periodic Vehicle Routing Problem", *Measurement*, Vol. 62, 2015, pp. 162-169.
18. Fallahi, A. E., Prins, C., and Wolfler Calvo, R., "A Memetic Algorithm and a Tabu Search for the Multi-Compartment Vehicle Routing Problem", *Computers & Operations Research*, Vol. 35, No. 5, 2008, pp. 1725-1741.
19. Chajakis, E. D. and Guignard, M., "Scheduling Deliveries in Vehicles with Multiple Compartments", *Journal of Global Optimization*, Vol. 26, No. 1, 2003, pp. 43-78.
20. Brown, G. G., Prins, C., and Wolfler Calvo, R., "Real-Time, Wide Area Dispatch of Mobil Tank Trucks", *Interfaces*, Vol. 17, No. 1, 1987, pp. 107-120.
21. Van der Bruggen, L., Gruson, R., and Salomon, M., "Reconsidering the Distribution Structure of Gasoline Products for a Large Oil Company", *European Journal of Operational Research*, Vol. 81, No. 3, 1995, pp. 460-473.
22. Avella, P., Boccia, M., and Sforza, A., "Solving a Fuel Delivery Problem by Heuristic and Exact Approaches", *European Journal of Operational Research*, Vol. 152, No. 1, 2004, pp. 170-179.
23. Reed, M., Yiannakou, A., and Evering, R., "An Ant Colony Algorithm for the Multi-Compartment Vehicle Routing Problem", *Applied Soft Computing*, Vol. 15, 2014, pp.

169-176.

24. Muyldermans, L., Ellis, C. J., Graves, G. W., and Ronen, D., “On the Benefits of Co-Collection: Experiments with a Multi-Compartment Vehicle Routing Algorithm”, *European Journal of Operational Research*, Vol. 206, No. 1, 2010, pp. 93-103.
25. Mendoza, J. E., Castanier, B., Guéret, C., Medaglia, A. L., and Velasco, N., “A Memetic Algorithm for the Multi-Compartment Vehicle Routing Problem with Stochastic Demands”, *Computers & Operations Research*, Vol. 37, No. 11, 2010, pp. 1886-1898.
26. Mladenović, N. and Hansen, P., “Variable Neighborhood Search”, *Computers & Operations Research*, Vol. 24, No. 11, 1997, pp. 1097-1100.
27. Lin, S. and Kernighan, B. W., “An Effective Heuristic Algorithm for the Traveling-Salesman Problem”, *Operational Research*, Vol. 21, No. 2, 1973, pp. 498-516.
28. Or, I., “Traveling Salesman-Type Combinatorial Problems and Their Relation to the Logistics of Regional Blood Banking”, PhD. Dissertation, Northwestern University, 1976.
29. Osman, I. H., “Metastrategy Simulated Annealing and Tabu Search Algorithms for the Vehicle Routing Problem”, *Annals of Operations Research*, Vol. 41, No. 4, 1993, pp. 421-451.
30. Potvin, J. Y. and Rousseau, J. M., “A Parallel Route Building Algorithm for the Vehicle Routing and Scheduling Problem with Time Windows”, *European Journal of Operational Research*, Vol. 66, No. 3, 1993, pp. 331-340.
31. Cordeau, J. F. and Laporte, G., “Tabu Search Heuristics for the Vehicle Routing Problem”, In Sharda, R. et al. (Eds.), *Metaheuristic Optimization via Memory and Evolution*, Springer, US, Vol. 30, 2005, pp. 145-163.
32. Golden, B. L., Wasil, E. A., Kelly, J. P., and Chao, I. M., “Metaheuristics in Vehicle Routing”, In Crainic, T. G. and Laporte, G. (Eds.), *Fleet Management and Logistics*, Kluwer, Boston, MA, 1998, pp. 35-36.
33. 林致瑄, 「應用 PSO 求解多艙種車輛路線問題」, 國立交通大學運輸與物流管理學系碩士論文, 民國 104 年。
34. Jianyong, J., Crainic, T. G., and Løkketangen, A., “A Cooperative Parallel Metaheuristic for the Capacitated Vehicle Routing Problem”, *Computers & Operations Research*, Vol. 44, 2014, pp. 33-41.

## 附錄：兩題 MCVRP 最新文獻最佳解之結果

|                                         |                                                                            |                                  |  |                                  |        |
|-----------------------------------------|----------------------------------------------------------------------------|----------------------------------|--|----------------------------------|--------|
| Problem instance: Set 2_Vrpnc9_no split |                                                                            | Best known solution cost: 1240.5 |  | Our result cost: 1181.1 (-4.79%) |        |
| No. of customers: 150                   | Capacity (a/b): 71.71/128.29                                               | Route Duration Constraint: 200   |  | Customer service time: 10        |        |
| Route #                                 | Route sequence                                                             |                                  |  | Loading (a/b)                    | Cost   |
| 1                                       | 0 - 132 - 1 - 122 - 30 - 131 - 32 - 90 - 108 - 10 - 70 - 101 - 69 - 0      |                                  |  | 48.10/97.08                      | 79.74  |
| 2                                       | 0 - 137 - 144 - 57 - 15 - 43 - 142 - 42 - 87 - 13 - 0                      |                                  |  | 41.00/73.24                      | 75.35  |
| 3                                       | 0 - 105 - 40 - 73 - 74 - 133 - 22 - 41 - 145 - 115 - 2 - 58 - 53 - 0       |                                  |  | 66.10/86.08                      | 64.72  |
| 4                                       | 0 - 28 - 12 - 150 - 80 - 68 - 121 - 29 - 129 - 3 - 77 - 116 - 76 - 111 - 0 |                                  |  | 57.70/126.50                     | 69.82  |
| 5                                       | 0 - 47 - 36 - 143 - 49 - 64 - 63 - 126 - 31 - 0                            |                                  |  | 43.80/82.33                      | 112.65 |
| 6                                       | 0 - 51 - 71 - 65 - 136 - 35 - 135 - 34 - 78 - 79 - 0                       |                                  |  | 48.00/68.08                      | 109.08 |
| 7                                       | 0 - 18 - 83 - 114 - 8 - 125 - 45 - 46 - 124 - 48 - 82 - 146 - 0            |                                  |  | 57.60/107.50                     | 89.13  |
| 8                                       | 0 - 26 - 149 - 54 - 130 - 55 - 25 - 24 - 134 - 109 - 138 - 0               |                                  |  | 45.30/89.83                      | 81.84  |
| 9                                       | 0 - 52 - 106 - 7 - 123 - 19 - 107 - 11 - 62 - 148 - 88 - 127 - 27 - 0      |                                  |  | 64.20/105.00                     | 74.60  |
| 10                                      | 0 - 110 - 4 - 139 - 39 - 67 - 23 - 56 - 75 - 72 - 21 - 0                   |                                  |  | 65.40/117.59                     | 96.14  |
| 11                                      | 0 - 147 - 104 - 99 - 5 - 61 - 16 - 141 - 44 - 119 - 14 - 97 - 117 - 0      |                                  |  | 69.20/120.84                     | 79.96  |
| 12                                      | 0 - 50 - 102 - 33 - 81 - 120 - 9 - 103 - 66 - 128 - 20 - 0                 |                                  |  | 69.70/118.26                     | 92.35  |
| 13                                      | 0 - 112 - 95 - 92 - 98 - 37 - 100 - 91 - 85 - 93 - 59 - 96 - 6 - 0         |                                  |  | 65.10/127.00                     | 56.75  |
| 14                                      | 0 - 94 - 38 - 140 - 86 - 113 - 17 - 84 - 118 - 60 - 89 - 0                 |                                  |  | 60.70/115.42                     | 99.03  |



|                                      |                                               |                                 |  |                                 |       |
|--------------------------------------|-----------------------------------------------|---------------------------------|--|---------------------------------|-------|
| Problem instance: Set 2_Vrpnc7_split |                                               | Best known solution cost: 952.6 |  | Our result cost: 926.4 (-2.75%) |       |
| No. of customers: 75                 |                                               | Capacity (a/b): 50.11/89.89     |  | Route Duration Constraint: 160  |       |
|                                      |                                               |                                 |  | Customer service time: 10       |       |
| Route #                              |                                               | Route sequence                  |  | Loading (a/b)                   |       |
|                                      |                                               |                                 |  | Cost                            |       |
| 1                                    | 0 - 6 - 33 - 43 - 41 - 56 - 23 - 63 - 0       |                                 |  | 34.40/87.75                     | 83.85 |
| 2                                    | 0 - 73 - 1 - 42 - 64 - 22 - 62 - 68 - 0       |                                 |  | 38.50/64.50                     | 89.84 |
| 3                                    | 0 - 67 - 0                                    |                                 |  | 10.00/20.00                     | 10.77 |
| 4                                    | 0 - 8 - 19 - 59 - 14 - 35 - 7(b) - 26 - 0     |                                 |  | 50.00/71.50                     | 82.78 |
| 5                                    | 0 - 5 - 37 - 20 - 70 - 60 - 71 - 0            |                                 |  | 28.60/55.50                     | 93.55 |
| 6                                    | 0 - 7(a) - 53 - 11 - 66 - 65 - 38 - 0         |                                 |  | 48.90/87.59                     | 77.23 |
| 7                                    | 0 - 32 - 50 - 18 - 55 - 25 - 9 - 0            |                                 |  | 42.50/70.58                     | 92.97 |
| 8                                    | 0 - 34 - 52 - 27 - 15 - 57 - 13 - 54 - 46 - 0 |                                 |  | 47.80/84.33                     | 78.61 |
| 9                                    | 0 - 2 - 28 - 61 - 21 - 74 - 30 - 0            |                                 |  | 43.30/86.75                     | 74.38 |
| 10                                   | 0 - 17 - 44 - 3 - 24 - 49 - 16 - 51 - 75 - 0  |                                 |  | 48.20/82.99                     | 76.87 |
| 11                                   | 0 - 4 - 45 - 29 - 36 - 69 - 47 - 48 - 0       |                                 |  | 46.70/76.41                     | 79.89 |
| 12                                   | 0 - 12 - 72 - 58 - 10 - 31 - 39 - 40 - 0      |                                 |  | 49.60/88.42                     | 85.71 |

