

# 演算法

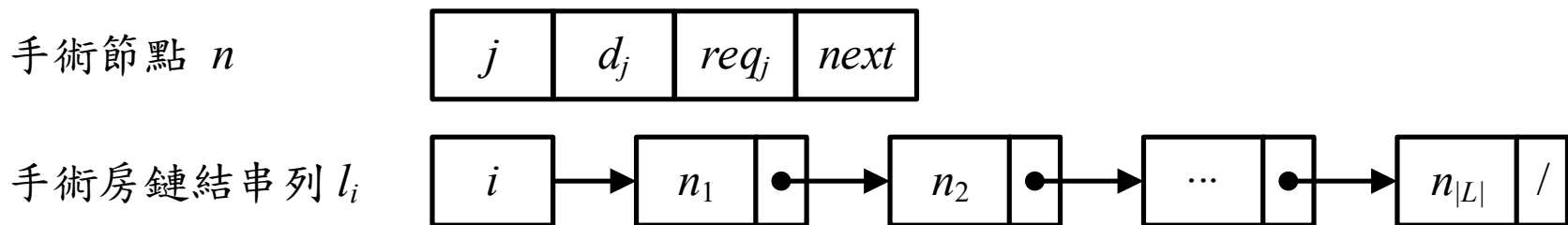
日期：0821

報告：王昱琪

# 資料結構設計：鏈結串列 (Linked List)

## 設計方法：

- 每間手術房  $i$  都有其對應的一鏈結串列集合  $L$ ，其中每一個鏈結串列由數個鏈結節點  $n_1, n_2, \dots, n_{|L|}$  所組成，每一鏈結節點由以下屬性組成：
  - 手術編號  $j$ ；
  - 手術時間  $d_j$ ；
  - 是否需特殊房  $req_j$ ，表示手術  $j$  是否需要特殊手術房，若需要則為 1，否則為 0；以及
  - 鏈結指標  $next$ 。



## 設計理由：

- 在手術排程問題中，每間手術房內需安排多臺手術，且手術順序經常需要調動。相較於一般陣列結構，鏈結串列在資料結構上具備顯著優勢：(1) 插入與刪除操作效率高，適應頻繁的排程調整需求；(2) 能清楚表達序列關係，明確呈現手術的先後順序；以及 (3) 操作彈性高，能有效支援排程的擾動與即時調整。因此，鏈結串列特別適合用於手術房內的排程管理，可有效解決傳統陣列結構在頻繁操作時面臨的時間複雜度問題。

# 成本函數

## 設計方法：

- 在此問題中預期達到最小化手術房加班與超時成本以及平衡各手術房使用時間，因此將最小化的目標函數建模如下：

$$\min W = \sum_{i \in I} (g_i z_i + h_i o_i) + \sum_{i \in I} \left( \left| s_i - \frac{1}{|I|} \times \sum_{i \in I} s_i \right| \right)$$

- 其中，
  - $I$ ：手術房集合，同時以  $|I|$  表示其集合數量；
  - $z_i$ ：手術房  $i$  的加班期間；
  - $o_i$ ：手術房  $i$  的超時期間；
  - $s_i$ ：手術房  $i$  的實際使用期間；以及
  - $g_i$ 、 $h_i$ ：超時成本與加班成本。

## 設計理由：

- 此目標函數設計旨在優化手術排程的整體效益，以兩個面向來討論成本控制與人力管理：(1) 成本控制面向，前半部分考量實務中手術房加班帶來的加班成本，以及超時產生的額外成本，直接反映營運的經濟負擔；以及 (2) 人力管理面向，後半部分引入平衡項讓整體排程更均衡，避免某些手術房過度集中負擔而其他手術房閒置，進一步提升醫護人員同時下班的可能性，減少人力等待與協作延遲。

# 選用演算法：模擬退火 (Simulated Annealing, SA)

## 設計方法：

1. 初始化溫度  $T$ 。
2. 讀入未排程前的資料當初始解  $X$ ，並計算其目標函數值  $E(X)$ 。
3. 根據擾動設定產生鄰近解  $Y$ ，並重新計算其目標值  $E(Y)$ 。若違反特殊手術房需求，則重新產生新解  $Y$ 。
4. 若  $Y$  優於目前最佳解，則直接接受，並令  $X=Y$  更新目前最佳解。
5. 若  $Y$  較差，則以下列機率接受： $P=\exp(-\Delta E/T)$ ，其中  $\Delta E=E(Y)-E(X)$ 。若接受，則仍令  $X=Y$ 。
6. 依冷卻策略降低溫度。
7. 重複步驟 2–6，直到滿足終止條件為止。

## 設計理由：

- 手術排程問題屬 NP-hard，傳統的整數規劃方法在大規模情境下計算成本過高，難以於臨床即時應用。  
SA 透過溫度控制與機率接受機制，在初期允許較差解以跳脫局部最佳，並逐步收斂至高品質解。

## 文獻：

- Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by Simulated Annealing. *Science*, 220(4598), 671–680. <https://doi.org/10.1126/science.220.4598.671>

# 整體演算法設計

- 初始溫度設定：使用二分搜尋法
- 擾動方式：
  - 擾動方式：擾動策略，根據溫度導引，多擾動策略的自適應切換。
  - 擾動方式：擾動次數，根據問題規模。
- 冷卻策略：根據問題規模

# 初始溫度設定：使用二分搜尋法

## 設計方法：

- 在演算法開始前，透過二分搜尋機制動態設定初始溫度  $T_0$ ，其操作流程如下：
  - 初始設定區間：設置搜尋範圍為  $[LB, UB]$  (待排手術數量\*手術房間數量)。
  - 測試方式：每次取目前區間的中位數作為測試溫度  $T_{test}$ ，執行演算法一輪。
  - 計算接受率= $move/iter$ ，其中， $move$  表示被接受的鄰解次數， $iter$  為總嘗試次數。
  - 調整區間：若接受率恰為所期望接受率，則將  $T_{test}$  定為初始溫度  $T_0$ ；若接受率低於期望，則將搜尋區間縮至下半段；若接受率高於期望，則將搜尋區間縮至上半段。
  - 重複搜尋：持續進行上述流程，直到找到最適合的初始溫度。

## 設計理由：

- 模擬退火的表現高度依賴初始溫度設定。若初溫過低，會導致早期幾乎無鄰解被接受，使搜尋停留在初始區域；若過高，則可能導致接受過多不良解而擴散過度。Lu et al. (2021) 指出，為了讓解空間在早期能有效展開，需讓接受率維持在一定範圍。因此使用二分搜尋法自動調整初始溫度，不但省去手動調參的困難，也能針對不同實例的特性設定最適起點，提升演算法的穩定性與適應性。

## 文獻：

- Lu, Z., Zhou, Y., & Hao, J.-K. (2022). A Hybrid Evolutionary Algorithm for the Clique Partitioning Problem. *IEEE Transactions on Cybernetics*, 52(9), 9391–9403. <https://doi.org/10.1109/tcyb.2021.3051243>

# 擾動方式：擾動次數，根據問題規模

## 設計方法：

- 擾動次數  $t_{greedy}$  依問題規模自動調整，設定為： $t_{greedy} = \beta \times n$
- 其中，
  - $\beta$ ：調整係數；以及
  - $n$ ：問題規模 (待排手術數量\*手術房間數量)。

## 設計理由：

- 擾動次數為演算法在每層溫度中探索鄰域解的關鍵設定。若固定次數，容易出現以下問題：在大規模排程中探索不足，導致過早收斂；在小規模問題中則可能造成不必要的計算資源浪費。手術數量不為固定數值，因此參考 Yuan et al. (2022) 所提方法，依據問題規模動態調整擾動次數，提升整體搜尋穩定性與彈性，使演算法更能適應不同規模之實際應用場景。

## 文獻：

- Yuan, Y., Tole, K., Ni, F., He, K., Xiong, Z., & Liu, J. (2022). Adaptive simulated annealing with greedy search for the circle bin packing problem. *Computers & Operations Research*, 144, 105826.  
<https://doi.org/10.1016/j.cor.2022.105826>

# 擾動方式：擾動策略，根據溫度導引，多擾動策略的自適應切換

## 設計方法：

- 擾動時，依據當前溫度階段決定使用的擾動策略與派遣規則。
- 整體流程分為三階段，透過一般化溫度選擇：

$$T_{norm} = \frac{T - T_{min}}{T_0 - T_{min}}$$

- 高溫階段：若滿足  $T_{norm} > T_H$  優先挑選執行時間長的手術，進行插入或交換操作，使手術跨房移動。藉此大幅改變房間的負載結構，以達到全域重組與跳脫局部最佳解的目的。
- 中溫階段：若滿足  $T_L < T_{norm} \leq T_H$  轉為選擇執行時間短的手術，進行插入或交換操作，調整局部時間配置與房間壓力，此階段著重於細部微調與解的細化。
- 低溫階段：若滿足  $T_{norm} \leq T_L$  且目前有手術房的實際使用期間仍到達設定的超時期間，則隨機抽選若干手術使所有手術房實際使用時間降至超時期間以下，並將抽出的手術以短刀優先與長刀優先兩種派遣規則重新安排，尋找更合適的放置位置。

## 設計理由：

- 在手術排程中，超時手術會造成嚴重的營運衝擊與醫療風險，因此必須盡量避免發生，同時持續改善整體解的品質。現行擾動方式多採單一策略一次用到底的做法，未能充分利用模擬退火法中溫度變化的特性。實際上應根據當前溫度階段調整擾動方式：(1) 高溫階段偏向探索，需要擾動幅度較大的方式，透過長時間手術的跨房移動達到全域重組，快速跳脫局部最佳解；(2) 中溫階段平衡探索與開發，改以短時間手術進行中等幅度擾動，在維持解結構穩定的前提下進行細部微調；以及 (3) 低溫階段專注開發，採用小幅度且針對性的擾動方式，集中處理明確的超時問題點，確保最終解的可行性與品質。此分階段擾動策略能有效提升演算法的收斂效率與解的穩定性。



# 冷卻策略：根據問題規模

## 設計方法：

- 採用自適應冷卻係數設計，於每一階段根據右式遞減溫度： $T_{k+1}=t_{cool}\cdot T_k$
- 其中，
  - $T_k$ ：第  $k$  次迭代的溫度；以及
  - $t_{cool}$ ：冷卻係數，控制溫度下降速度，數值越大降溫越慢： $t_{cool}=\frac{\alpha\times\sqrt{n}-1}{\alpha\times\sqrt{n}}$ 
    - ◆ 其中，
      - $\alpha$ ：調整係數；以及
      - $n$ ：問題規模 (待排手術數量\*手術房間數量)。

## 設計理由：

- 自適應冷卻可依排程規模調整降溫速度，避免大規模問題過早收斂、小規模問題浪費資源。實務上，手術數量每日波動大，固定冷卻率易造成效率與品質失衡，故使用 Yuan et al. (2022) 所提方法，依手術數量調整冷卻係數，提升穩定性與彈性。

## 文獻：

- Yuan, Y., Tole, K., Ni, F., He, K., Xiong, Z., & Liu, J. (2022). Adaptive simulated annealing with greedy search for the circle bin packing problem. *Computers & Operations Research*, 144, 105826.  
<https://doi.org/10.1016/j.cor.2022.105826>