# Fujitsu Digital Annealer（DA）約束功能完整整理

## 1) DA 的核心輸入：QUBO

DA 的核心輸入是 QUBO（Quadratic Unconstrained Binary Optimization），也就是把所有目標與約束都轉成二次二元多項式的能量函數：

E(x) = ∑(i<j) J\_ij \* x\_i \* x\_j + ∑(i) h\_i \* x\_i + c （官方使用手冊，富士通）

## 2) HOBO → QUBO 自動轉換

提供 hobo2qubo API，把更高次的項自動降次成 QUBO，等同於幫你做標準的「四次以上→二次」的量化／引入輔助變數流程（內含係數縮放與整數化的機制）。（富士通）

## 3) 係數自動縮放與整數化（Scaling & Rounding）

若 QUBO 係數超過硬體可表示範圍或非整數，求解器會自動縮放並四捨五入到 DA 可接受的整數範圍（包含 64-bit/76-bit 限制與建議範圍）。（富士通）

以上代表：即使你來自 CP-SAT／MIP／高次懲罰設計，DA 端已提供把模型「清理」成可運行 QUBO 的實務工具鏈。

## 4) 直接「內建處理」的典型約束

DA（第 3、4 代 DA3c/DA4）透過官方與主流 SDK（如 Fixstars Amplify、Strangeworks 參數層）對幾種常見結構化約束提供一等公民（first-class）的支援：

- One-hot 類（唯一選擇）：
 • One-way one-hot：一組變數中「恰好一個」或「至多一個」為 1。
 • Two-way one-hot：矩陣式（每列、每行恰好一個），典型於指派、排程、Sudoku。

- 線性不等式／等式：
 • 一次線性不等式可作為「分離的約束」處理，穩定性高。
 • 等式約束可直接宣告，必要時退回懲罰式。

- 固定／引導設定：
 • fixed\_config 固定變數值。
 • guidance\_config 提供初始解，引導搜尋。

## 5) SDK / 參數層指定方式

- Fixstars Amplify：提供 equal\_to、less\_equal、greater\_equal、one\_hot 等 API；支援 one-way/two-way one-hot 與 inequality 分離。
- Strangeworks：提供 one\_way\_one\_hot\_groups、two\_way\_one\_hot\_groups、penalty\_coef、penalty\_auto\_mode、fixed\_config 等參數。

## 6) 模型設計對應與實例

- 指派／排班／VRP：使用 two-way one-hot 確保每人每班唯一。
- 圖分割／社群偵測：用 one-hot 分群，群大小再加 inequality。
- 排程資源上限：使用 inequality ∑(x\_it) ≤ cap\_t。
- 組合化學設計：每位點 one-hot，或多位點 two-way one-hot。

## 7) 懲罰與約束互動

多數 QUBO 求解依靠懲罰項逼可行解；但 DA 的 one-hot 與線性 constraint-aware 處理結合 penalty 自動化（penalty\_auto\_mode, penalty\_inc\_rate），減少人工調參。文獻指出「不等式分離」能顯著減少微調懲罰時間。

## 8) 進階與限制提醒

- DA 本質仍是 QUBO 求解器，高次或整數／實數變數需經 HOBO→QUBO 或編碼方式轉換。
- 係數範圍需注意，過大懲罰會被縮放取整，導致能量地形失真；建議使用 penalty\_auto\_mode 與官方 scaling 規範。