

# 系統分析與設計 期中報告

## 工廠生產排程規劃

第一組

許哲榮、吳泊諄、王梓旭、陳柄瑞

April 2021

### 1 前言

在多數工廠中，有多條生產線平行運作，同時生產不同產品，產品間有不同的生產流程以及所需製造時間。現今傳統產業中，多數工廠採用人工經驗進行排程 (scheduling)，因此不一定能得到最佳 (或近似最佳) 的結果 (例如: 最少延遲訂單數、最早完成所有訂單)。因此，若有系統使用演算法排程機台生產規劃將有助於提升生產效率。

#### 1.1 目的

某食品公司 (以下簡稱 B 公司) 欲將其生產線排程系統資訊化，希望增加排程系統的效率與效能。而本專案小組即以實作此資訊化排程系統為目標。在工廠中機台分為強機台與弱機台的情況下，以最小化 makespan 為目標。

#### 1.2 範圍

本專案負責之工作範圍如下：

- 建立資料庫儲存客戶歷史排程資料。
- 設計啟發式演算法找出近似最佳的數個排序方式。
- 提供本機網頁介面供客戶操作。

- 提供客戶調整排程設定功能。

### 1.3 技術性字詞

往後文件中將提到數個技術性字詞，於此將其定義：

- **Makespan**：即工廠所有機台中，最晚完成所有工作的時間。亦即工廠可以下班的時間。
- **強機台**：可以進行所有加工工序之機台。
- **弱機台**：僅可以進行特定加工工序之機台。

### 1.4 版本

B 公司目前無資訊系統，因此本系統將成為 B 公司第一個生產排程系統。

## 2 遭遇問題

現況	問題
公司內部資料未資訊化，皆為紙本紀錄。	<ul style="list-style-type: none"> <li>• 資料常因字跡潦草而難以辨識。</li> <li>• 資料保存不易，系統化保存成本高，容易遺失。</li> </ul>
生產排程方式依靠人工經驗。	<ul style="list-style-type: none"> <li>• 排程過程耗時費力。</li> <li>• 排程成效不一定足夠好，以致工廠人員需要加班而造成額外人力成本。</li> </ul>
工廠部分機台功能不全，無法負擔所有工序需求。	<ul style="list-style-type: none"> <li>• 部分機台功能不完全，導致特定商品需要移至其他機台完成，增加排程複雜度，降低排程成效。</li> </ul>

## 3 系統描述

### 3.1 環境需求

- 作業系統採用 Windows。執行後可經由瀏覽器 (Browser)，例如：Microsoft Edge、Firefox 或 Chrome 等方式開啟應用程式。
- 能夠支援輸入 Excel 格式資料，並規範好資訊化的檔案格式，製作 Excel 模板。

### 3.2 介面需求

- 於網站完成後，進行使用者端及操作介面等測試，各項功能均應符合需求。各式功能整理並排版，放置於網站首頁上，使各式功能一目了然。
- 排程檔案上傳後，可以檢視上傳之檔案內容，以確認檔案是否正確。
- 對於任何資料維護，於刪除前有確認視窗提醒以確認。
- 於 Microsoft Edge、Firefox 或 Chrome 等瀏覽器上皆能正常運作。
- 不須進行安裝，只需要將檔案解壓縮後並執行，即可使用。

### 3.3 功能需求

- 選擇日期，並上傳排程檔案以新建排程。
- 可以將排程結果下載，以利後續使用。
- 查看以及刪除歷史排程。
- 查看以及修改系統設定。
- 可以取消目前動作，回到上一頁。
- 上傳資料後，檢查資料型態及必填欄位是否正確。

### 3.4 功能說明

- (a) 系統、網站以及資料庫等皆架設於本機，整個網站運行皆由本機進行。程式語言以 Python 為主，並以 Django 實作後端資料庫，輔以 HTML 以及 CSS 設計網站架構。
- (b) 排程規劃主要以啟發式演算法為主，同時也會使用 Gurobi Optimizer 進行比較，比較最佳解以及啟發式演算法找到的可行解之間的差距，以此為評估指標找尋接近最佳解的可行解。
- (c) 為方便工廠人員查詢，需要將歷史排程結果儲存，並可以在網頁上瀏覽和下載，也可刪除。
- (d) 為因應工廠不同狀況，需要系統設定，用以機台數、開始時間等設定。

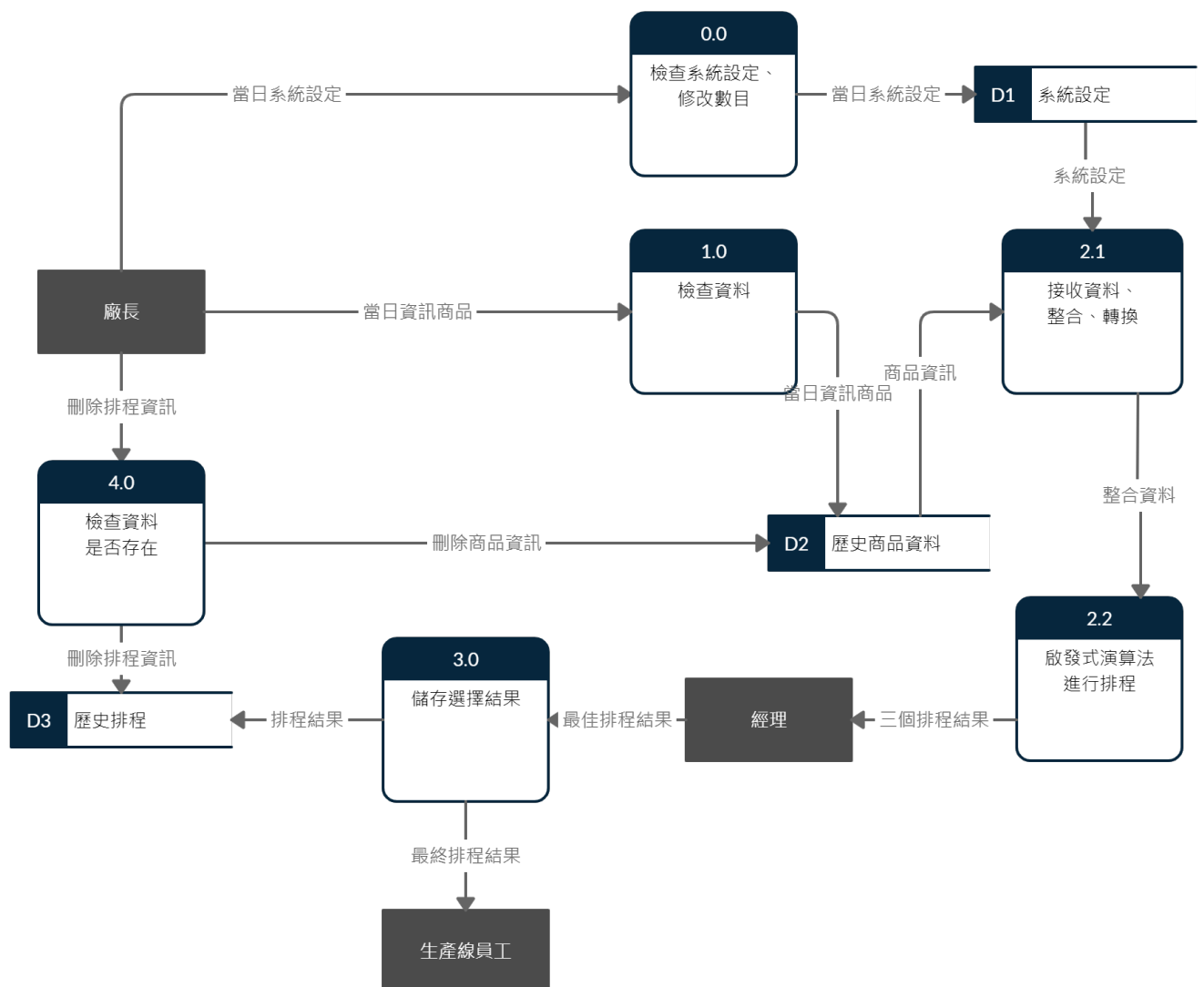


Figure 1: Data Flow Diagram

### 3.5 系統效益

(a) 解決資料保存與辨識問題

原先紙本資料不易保存且可讀性不高，資訊化後將資料依格式存至電腦可解決保存問題並增加資料可讀性。

(b) 解決排程耗時費力的情況

資訊化的系統根據輸入的資料產出排程結果，過程快速且只需簡單地上傳資料，解決原先耗時費力的情況，降低排程時間與人力。

(c) 解決排程成效低落問題

原先人工排程成效無法保證，但系統依據效能評估過的演算法排程，將確保排程結果的成效。解決排程成效低落問題，降低員工加班機會。

(d) 因應工廠部分機台功能低落

工廠中部分機台功能不完全，使排程複雜化，降低排程成效。資訊化系統將代替人力完成排程，同時確保成效，降低工廠部分機台功能低落造成的影響。

## 4 評估分析

(a) 技術能力

團隊擁有演算法、前後端軟體開發人才，足以完成生產排程資訊系統，且本系統無需網路連線功能，開發時不用考慮資訊安全、隱私保護等問題。

(b) 設備資源

團隊成員皆有能應付演算法運算資源的電腦與開發環境，且 B 公司也有相關軟體設備。

(c) 開發時程

本系統規模不大，依據團隊成員軟體開發經驗，三個半月應能如期完工。

(d) 經濟考量

本專案採取結算報酬制，系統完成後，B 公司會決定是否購買本系統，若決定購買，則團隊將依據系統完成度與 B 公司議價，因此開發過程中，團隊無訂金壓力，成員將考量自身能力與時間決定投入之精力。

## 5 其他考量

### (a) 可能風險

因 B 公司未給付相關訂金，系統完成後若 B 公司不滿意決定不購買，可能無法取得任何金錢上之酬勞。

### (b) 相關假設

工廠環境複雜，影響生產之變因眾多，若 B 公司無法提供準確的排程限制，團隊開發之演算法可能無法應付現實狀況。

## 6 時程規劃

預計時間	工作內容
三月	訪談：與工廠負責人確認問題、需求與目標，並了解工廠運行現況。
四月	實作：比較啟發式演算法 (Heuristic Algorithm) 與 Gurobi Optimizer 最佳解之差距。
五月	開發：建設前端網頁與後端資料庫之系統，供日後工廠負責人使用。
六月初	驗收：交由工廠負責人驗收並講解系統使用方法，確認是否達成目標。
六月中	修正：對於工廠負責人可能提出之建議，進行改善與微調。

## 7 工作分配

組員	工作內容
許哲榮	啟發式演算法、前端系統開發
吳泊諄	啟發式演算法、後端系統開發
王梓旭	前後端系統整合、軟體測試
陳柄瑞	訪談與聯絡、啟發式演算法

## 8 需求訪談

在此份計劃書中，我們的訪談對象為 B 食品公司的經理。B 公司屬傳統食品公司，有多種產品需在多台機台生產。由於距離因素，我們的訪談採取線上進行。透過事前取得的資料，我們準備多項問題，以向經理確認系統需求以及公司目前的規劃方法。整體來說，訪談的過程相對順利，我們獲得更詳細的需求以及明確的商品生產流程與限制，對計劃有更透徹的了解。

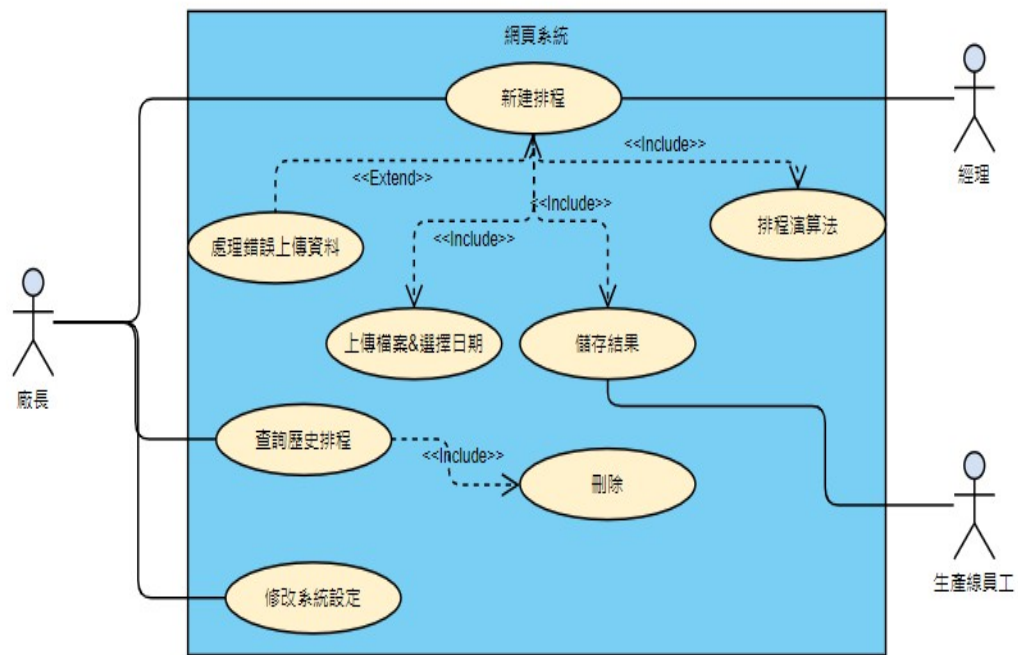


Figure 2: Use Case Diagram

## 9 附錄：訪談紀錄

- 時間：110 年 3 月 19 日早上 9 點至 11 點
- 訪談對象：B 食品公司的經理
- 地點：線上
- 主持：吳泊諄
- 出席：許哲榮、王梓旭、吳泊諄、陳炳瑞
- 紀錄：許哲榮

### (a) 生產線以及排程需求與限制

- Q: 當前生產商品的排程方法為何？  
A: 每日早上開工前，我會依據過去經驗手動規劃排程。
- Q: 工廠內的機台有幾台？  
A: 一共五台，有一台為弱機台，與四台強機台。
- Q: 一車商品的工序能否於不同機台先後運作？  
A: 可以，但最多只能換機台一次，要不然會花很多時間。
- Q: 同訂單的不同車可否平行運作？  
A: 同訂單的不同車可以平行運作。每車可以分開做，不同車可以同時在不同機台生產。
- Q: 商品工序中的清潔、拉出時間為何？  
A: 清潔時間：會再去詢問。拉出時間：固定某幾個工序需要將商品從機台拉出人工處理，如煙燻步驟需要拉出翻面。

### (b) 系統需求

- Q: 我們的系統會建在個人電腦中，無法進行遠端連線。想問屆時操作的電腦系統是什麼，平常使用哪個瀏覽器？  
A: 電腦是 Windows 系統，Win 10 版本，瀏覽器是用 Chrome 或 Microsoft Edge。
- Q: 每日將商品資訊輸入系統的方式為何？會希望上傳固定格式的商品檔案，還是在網頁上填入商品資訊？



A: 會希望每天上傳 Excel 檔，因為有時候會改良製程，有時不同天、同商品的工序、時間長度會不同。

(c) 經理提問

- Q: 系統完成後，輸出格式為何？

A: 我們會將結果表格化並以圖示化呈現。圖示化的部分會畫成甘特圖，可以初步一覽排程結果。

- Q: 最佳解（窮舉法）與我們自己設計的演算法結果有所差別，為何我們不採用窮舉法？

A: 由於窮舉法會需要花費大量時間（至少三個小時），因此將以更直觀設計的演算法代替。雖然可能不會產生最好的結果，但透過多種演算法組合，可以找到不錯的結果。