Metaheuristics for the green scheduling of a two-machine flowshop considering energy consumption

Perpetual Naomi Avornor [1] GJ Le Roux [2] Isabelle Nieuwoudt [2]

[1]AIMS South Africa [2]Stellenbosch University, South Africa

Abstract

This study tackles the challenge of optimising energy consumption in a two-machine flowshop scheduling problem. A mixed-integer linear programming model is used to minimise makespan, energy consumption, processing, and idle times. Two single-solution based metaheuristics namely tabu search and simulated annealing were applied to solve this NP-hard problem, with benchmarks showing performance from both methods. The findings provide insights into achieving efficient, sustainable manufacturing processes through optimised scheduling.

Motivation

- **Rising Energy Costs:** Manufacturing industries are facing increasing energy costs, necessitating more energy-efficient production scheduling.
- **Environmental Responsibility:** With stricter environmental regulations, there is a growing demand for sustainable manufacturing practices.
- **Industry Need:** There's a critical need for practical scheduling solutions that balance energy use and production efficiency to meet both economic and environmental goals.

Objectives

1. **Optimise Energy Use:** Minimise energy consumption while maintaining production efficiency in a two-machine flowshop.
2. **Multi-Objective Optimization:** Simultaneously optimise makespan, processing time, idle time, and energy consumption.
3. **Evaluate Metaheuristics:** Compare the performance of tabu search and simulated annealing for solving this NP-hard problem across different problem scales.

Methodology

- **Problem Formulation:**
  A Mixed-Integer Linear Programming (MILP) model is used, considering sequence-dependent setups, idle power, and discrete speed options.
- **Algorithms Used:**
  Tabu search: A memory-based local search method used to tackle complex combinatorial problems.
  Simulated annealing: A probabilistic search method that takes inspiration from metalworking process where material are heated and then cooled slowly to remove defect.

Mathematical Formulation

Objective Functions:

- **Minimise Makespan (g):**

$$g \geq c_{j2}$$

where c_{j2} is the completion time of job j on machine 2.

- **Minimise Energy Consumption (e):**

$$e = \sum_{j=1}^{n}\sum_{\ell=1}^{3}\sum_{m=1}^{2} u_{j\ell m}\left(\frac{p_{jm}}{60 r_\ell}\beta_\ell \rho_m\right) + \sum_{m=1}^{2}\frac{\gamma_m \rho_m}{60} t_m$$

where p_{jm} is the processing time, r_ℓ is the speed factor, β_ℓ and ρ_m are machine-specific parameters, and t_m is idle time on machine m.

- **Minimise Total Processing Time (q):**

$$q = \sum_{j=1}^{n}\sum_{\ell=1}^{3}\sum_{m=1}^{2}\frac{p_{jm}}{r_\ell}u_{j\ell m}$$

- **Minimise Idle Time (t):**

$$t_m = g - \sum_{j=1}^{n}\sum_{\ell=1}^{3}\frac{p_{jm}}{r_\ell}u_{j\ell m}$$

Decision Variables:

- $y_j \in \{0,1\}$: Binary variable indicating if job j is the first in the sequence.
- $x_{jk} \in \{0,1\}$: Binary variable for job sequencing, where 1 indicates that job j precedes job k.
- $u_{j\ell m} \in \{0,1\}$: Binary variable indicating the machine speed for job j on machine m.

Constraints:

- **Job Sequence Constraints:**

$$\sum_{j=1}^{n} y_j = 1 \quad \text{(only one job is the first)}$$

$$\sum_{k=1, k\neq j}^{n} x_{jk} = 1, \quad \sum_{j=1, j\neq k}^{n} x_{jk} = 1 \quad \text{(each job is assigned exactly once in the sequence)}$$

- **Machine Speed Constraints:**

$$\sum_{\ell=1}^{3} u_{j\ell m} = 1, \quad \forall j, m \quad \text{(each job is processed at exactly one speed on each machine)}$$

- **Completion Time Constraints:**

$$c_{j1} \geq \frac{p_{j1}}{r_\ell}u_{j\ell 1} + d_{jj1}y_j \quad \forall j$$

$$c_{j2} \geq c_{j1} + s_j + \frac{p_{j2}}{r_\ell}u_{j\ell 2} \quad \forall j$$

- **Setup Time and Sequence Constraints:**

$$L(1 - x_{jk}) + s_j \geq d_{jj2} - c_{j1}, \quad j \neq k$$

$$c_{j2} \geq c_{j1} + s_j + \frac{p_{j2}}{r_\ell}u_{j\ell 2}, \quad j = 1, \ldots, n$$

How the algorithms work

Tabu Search (TS)

- Iteratively improves the solution by exploring its neighborhood.
- Uses a *tabu list* to avoid revisiting solutions.
- Helps escape local minima with aspiration criteria.
- Repeats until a stopping condition is met.
- Returns the best solution.

Simulated Annealing (SA)

- Accepts worse solutions with decreasing probability over time using Metropolis criteria.
- Temperature decreases, reducing the likelihood of accepting worse solutions.
- Helps avoid local optima by exploring diverse solutions.
- Returns the best solution.

Experimental Results

Convergence Analysis

The convergence analysis was used to determine the maximum execution time for each algorithm. This demonstrates how the difficulty levels of benchmark behave over time.

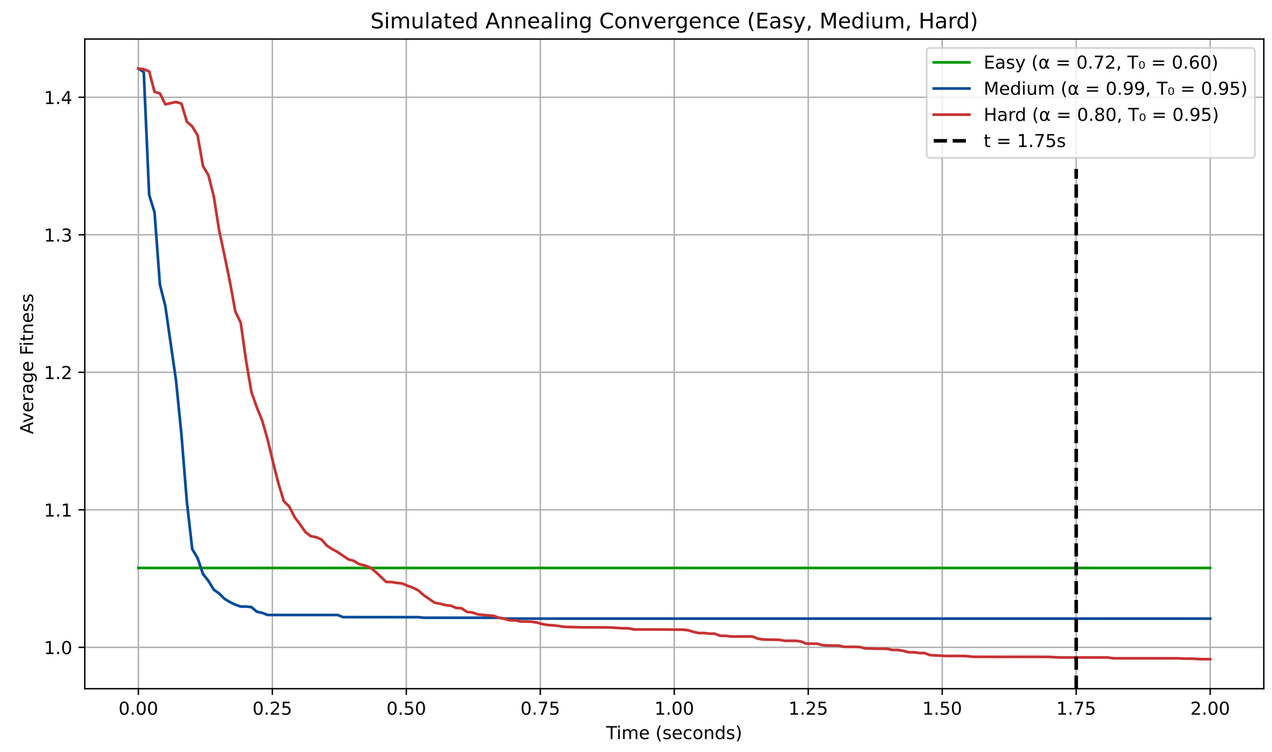

Figure 1. Convergence graphs for different problem difficulty levels.

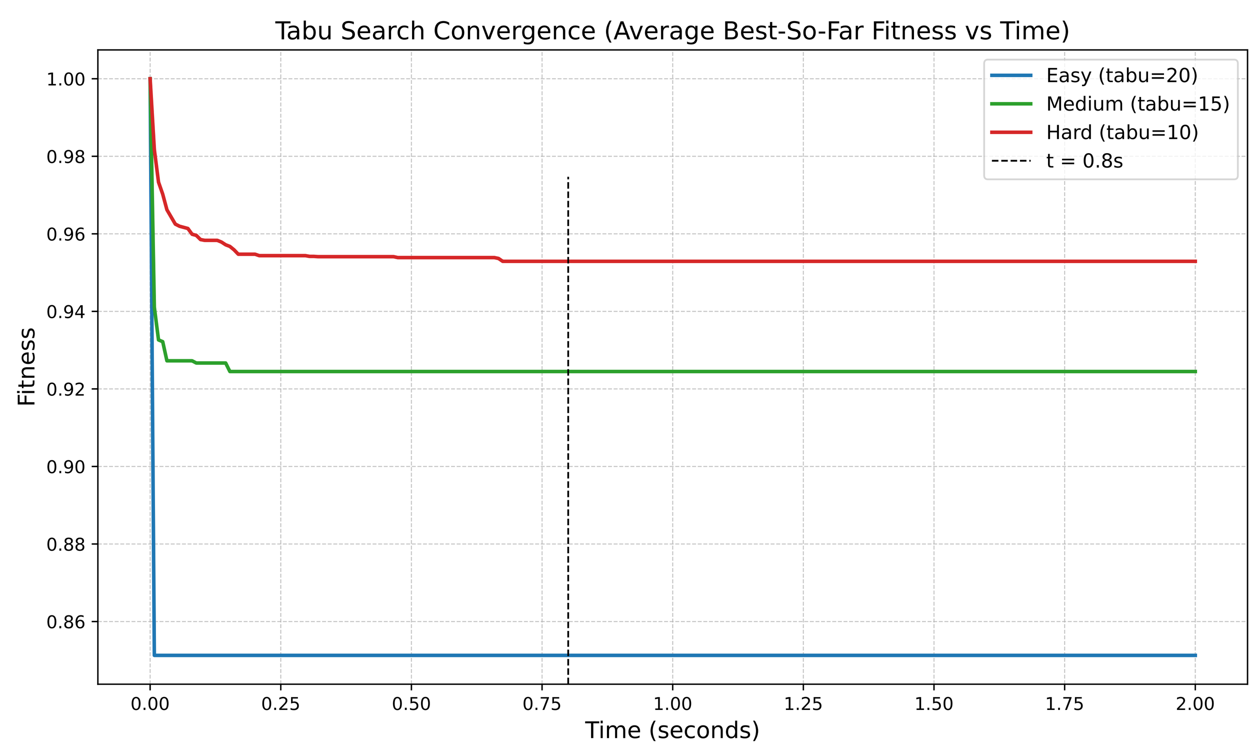

Figure 2. Convergence graph for SA parameters.

Algorithmic Comparisons

This section compares tabu search and simulated annealing in terms of solution quality and computational efficiency across different problem sizes.

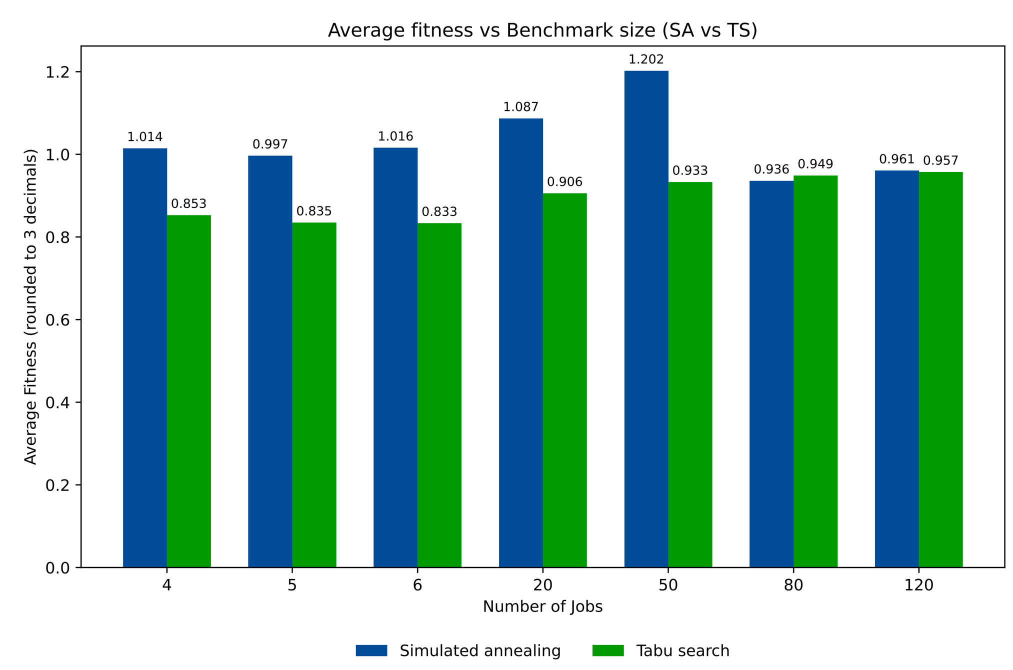

Figure 3. Comparison of solution quality between SA and TS.

Conclusion

This study demonstrates the effectiveness of tabu search and simulated annealing in solving energy-efficient scheduling problems in a two-machine flowshop. Both algorithms offer valuable trade-offs in terms of solution quality and computational efficiency, with TS excelling in smaller problems and SA handling larger instances effectively. The results highlight the potential of metaheuristics in optimising manufacturing processes for sustainability and efficiency.