

## Work scheduling optimization with operational constraints

### Optimasi penjadwalan kerja dengan kendala operasional

Ulva Widowati<sup>1</sup>, Ahmad Ilham, S.Kom.,M.Kom<sup>2</sup>

<sup>1</sup>Program Studi Informatika, Fakultas Teknik Universitas Muhammadiyah Semarang, Semarang, Indonesia

<sup>2</sup>Program Studi Informasi, Fakultas Teknik, Universitas Muhammadiyah Semarang, Semarang, Indonesia

#### Info Artikel

##### Riwayat Artikel:

Diterima 10, Desember, 2024  
Perbaikan 17, Januari, 2025  
Disetujui 30, Januari, 2025

##### Keywords:

Jadwal Kerja  
Efisien  
Matematis  
PuLP

#### ABSTRAK

Penjadwalan kerja yang efisien menjadi tantangan utama dalam pengelolaan tenaga kerja di perusahaan manufaktur. Paper ini menghadirkan solusi berbasis pemrograman linier untuk mengoptimalkan jadwal kerja 15 karyawan selama 7 hari, dengan tujuan memenuhi kebutuhan tenaga kerja harian sekaligus meminimalkan biaya operasional. Model matematis yang dirancang mempertimbangkan kendala jam kerja maksimal per karyawan dan jumlah tenaga kerja minimum harian. Implementasi dilakukan menggunakan Python dengan pustaka PuLP, yang menghasilkan jadwal kerja optimal dengan total biaya tenaga kerja minimal. Hasil analisis menunjukkan bahwa solusi ini tidak hanya memenuhi semua kendala operasional, tetapi juga memberikan fleksibilitas untuk diterapkan di berbagai konteks organisasi.

#### ABSTRACT

*Efficient work scheduling is a major challenge in manpower management in manufacturing companies. This paper presents a linear programming-based solution to optimize the work schedule of 15 employees for 7 days, with the aim of meeting daily manpower requirements while minimizing operational costs. The mathematical model designed considers the constraints of maximum working hours per employee and minimum daily manpower. The implementation was done using Python with the PuLP library, which resulted in an optimal work schedule with minimal total labor costs. The analysis results show that this solution not only satisfies all operational constraints, but also provides flexibility to be applied in various organizational contexts.*

*Ini adalah artikel akses terbuka di bawah lisensi CC BY-SA.*



#### Penulis Korespondensi:

Ulva Widowati  
Program Studi Informatika, Fakultas Teknik Universitas Muhammadiyah Semarang  
Alamat: Gedung FT-MIPA Lt. 7, Ruang 707, Jl.Kedungmundu Raya No.18, Semarang 50273, Indonesia  
Email: [ulvachan07@gmail.com](mailto:ulvachan07@gmail.com)

#### 1. PENDAHULUAN

Efisiensi operasional dalam manajemen tenaga kerja merupakan salah satu faktor kunci keberhasilan perusahaan, terutama di sektor manufaktur yang sangat bergantung pada optimalisasi sumber daya manusia. Dalam konteks ini, penjadwalan kerja yang baik tidak hanya memastikan kebutuhan operasional terpenuhi, tetapi juga berperan penting dalam mengendalikan biaya tenaga kerja, yang sering kali menjadi salah satu komponen terbesar dalam anggaran perusahaan.

Namun, penjadwalan kerja merupakan masalah kompleks yang dipengaruhi oleh berbagai kendala, seperti jumlah minimum karyawan yang harus tersedia setiap hari, batas maksimum jam kerja individu, serta

variasi upah tenaga kerja. Penyelesaian manual terhadap masalah ini cenderung memakan waktu dan rentan terhadap kesalahan, terutama dalam skala yang besar. Oleh karena itu, pendekatan berbasis metode kuantitatif seperti pemrograman linier menjadi solusi yang efektif.

Paper ini bertujuan untuk mengembangkan model matematis berbasis pemrograman linier guna menyelesaikan masalah penjadwalan kerja dengan kendala operasional. Model ini tidak hanya dirancang untuk memenuhi kebutuhan minimum tenaga kerja harian, tetapi juga untuk meminimalkan total biaya tenaga kerja. Dengan menggunakan Python dan pustaka optimasi seperti PuLP, implementasi komputasi dilakukan untuk menghasilkan jadwal kerja yang optimal.

Kontribusi utama penelitian ini adalah menyajikan metode terintegrasi yang dapat diterapkan dalam berbagai konteks industri untuk meningkatkan efisiensi penjadwalan kerja. Selain itu, paper ini juga memberikan analisis mendalam terhadap solusi yang dihasilkan, dengan mempertimbangkan efisiensi biaya dan fleksibilitas operasional sebagai parameter utama.

## 2. METODE

### 2.1. Model Matematis

Model matematis yang digunakan dalam penelitian ini adalah pendekatan pemrograman linier, yang dirancang untuk mengoptimalkan jadwal kerja karyawan dalam sebuah perusahaan manufaktur. Model ini bertujuan meminimalkan total biaya tenaga kerja sambil memastikan bahwa kebutuhan tenaga kerja harian terpenuhi sesuai kendala operasional yang ada.

#### 1. Variabel Keputusan

Variabel keputusan  $x_{i,j}$  didefinisikan sebagai jumlah jam kerja yang dijadwalkan untuk karyawan  $i$  pada hari  $j$ , dengan:

$$x_{i,j} \geq 0, \forall i \in \{1, 2, \dots, 15\}, j \in \{1, 2, \dots, 7\} \quad \forall i \in \{1, 2, \dots, 15\}, j \in \{1, 2, \dots, 7\}$$

Variabel ini mewakili keputusan utama dalam model, yaitu menentukan berapa jam kerja setiap karyawan setiap hari dalam satu minggu.

#### 2. Fungsi Objektif

Tujuan dari model ini adalah meminimalkan total biaya tenaga kerja selama 7 hari kerja. Fungsi objektif dinyatakan sebagai:

$$\text{Minimalkan } Z = \sum_{i=1}^{15} \sum_{j=1}^7 (\text{Upah per Jami} \cdot x_{i,j})$$

di mana Upah per Jami adalah biaya tenaga kerja per jam untuk karyawan  $i$ . Fungsi ini memastikan bahwa total biaya tenaga kerja diminimalkan, dengan tetap memenuhi kendala operasional yang ditentukan.

#### 3. Kendala Operasional

Model ini mencakup beberapa kendala utama yang harus dipenuhi:

##### a. Kebutuhan Minimum Tenaga Kerja Harian

Setiap hari, perusahaan membutuhkan setidaknya 8 karyawan yang bekerja untuk menjaga kelancaran operasi. Kendala ini dirumuskan sebagai:

$$\sum_{i=1}^{15} x_{i,j} \geq 8, \forall j \in \{1, 2, \dots, 7\}$$

Kendala ini memastikan bahwa jumlah total jam kerja yang dialokasikan setiap hari mencukupi kebutuhan operasional perusahaan.

##### b. Jam Kerja Maksimum per Karyawan

Setiap karyawan memiliki batas maksimum jam kerja sebesar 48 jam per minggu untuk mencegah kelelahan dan menjaga kepatuhan terhadap aturan ketenagakerjaan. Kendala ini dinyatakan sebagai:

$$\sum_{j=1}^7 x_{i,j} \leq 48, \forall i \in \{1, 2, \dots, 15\}$$

Kendala ini membatasi alokasi jam kerja per karyawan agar tidak melebihi batas yang ditentukan.

##### c. Jam Kerja Non-Negatif

Jam kerja yang dijadwalkan untuk setiap karyawan setiap hari harus berupa nilai nonnegatif:

$$x_{i,j} \geq 0, \forall i \in \{1, 2, \dots, 15\}, j \in \{1, 2, \dots, 7\}$$

Kendala ini diperlukan karena nilai negatif tidak memiliki makna dalam konteks jumlah jam kerja.

#### 4. Struktur Model Secara Keseluruhan

Model pemrograman linier secara keseluruhan dapat dirangkum sebagai berikut:

Fungsi Objektif:

Minimalkan  $Z = \sum_{i=1}^{15} \sum_{j=1}^7 (\text{Upah per Jami} \cdot x_{i,j})$   $Z =$

$\sum_{i=1}^{15} \sum_{j=1}^7 (\text{Upah per Jam})_i \cdot x_{i,j}$  Kendala:

1.  $\sum_{i=1}^{15} x_{i,j} \geq 8, \forall j \in \{1, 2, \dots, 7\}$   $\sum_{i=1}^{15} x_{i,j} \geq 8, \quad \forall j \in \{1, 2, \dots, 7\}$

2.  $\sum_{j=1}^7 x_{i,j} \leq 48, \forall i \in \{1, 2, \dots, 15\}$   $\sum_{j=1}^7 x_{i,j} \leq 48, \quad \forall i \in \{1, 2, \dots, 15\}$

$x_{i,j} \geq 0, \forall i \in \{1, 2, \dots, 15\}, j \in \{1, 2, \dots, 7\}$   $x_{i,j} \geq 0, \quad \forall i \in \{1, 2, \dots, 15\}, j \in \{1, 2, \dots, 7\}$ .

### 5. Penjelasan Konseptual

Model ini memberikan fleksibilitas dalam menentukan kombinasi alokasi jam kerja yang optimal, dengan mempertimbangkan biaya tenaga kerja yang berbeda-beda antar karyawan. Fungsi objektif bertindak sebagai parameter efisiensi biaya, sementara kendala memastikan jadwal yang dihasilkan tetap realistis dan operasional.

## 3. HASIL DAN PEMBAHASAN

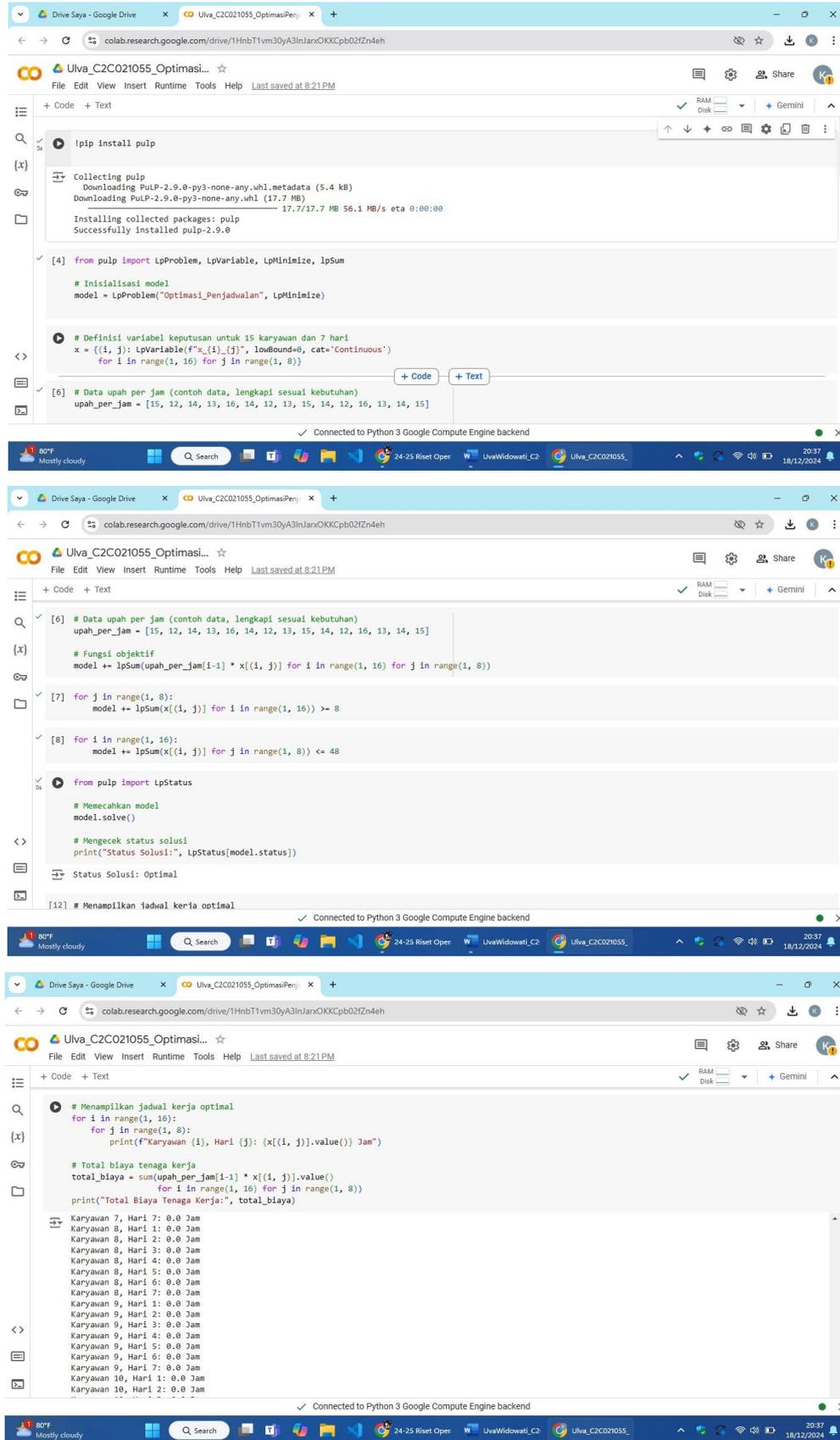
### 3.1. Implementasi Komputasi

Model matematis ini diimplementasikan menggunakan Python dengan pustaka optimasi seperti PuLP. Implementasi ini memungkinkan pengambilan keputusan yang cepat dan akurat, terutama dalam skenario dengan jumlah karyawan dan kendala yang lebih kompleks.

### 3.2. Implementasi Python

Implementasi model matematis dilakukan menggunakan bahasa pemrograman Python dengan pustaka PuLP, yang dirancang khusus untuk memecahkan masalah optimasi. Proses implementasi ini mencakup beberapa langkah utama, yaitu mendefinisikan variabel keputusan, fungsi objektif, kendala, dan memecahkan model dengan solver optimasi.

```
!pip install pulp
from pulp import LpProblem, LpVariable, LpMinimize, lpSum
# Inisialisasi model model = LpProblem("Optimasi_Penjadwalan", LpMinimize)
# Definisi variabel keputusan untuk 15 karyawan dan 7 hari x = {(i, j): LpVariable(f"x_{i}_{j}", lowBound=0,
    cat='Continuous') for i in range(1, 16) for j in range(1, 8)}
# Data upah per jam (contoh data, lengkapi sesuai kebutuhan) upah_per_jam = [15, 12, 14, 13, 16, 14, 12, 13,
    15, 14, 12, 16, 13, 14, 15] # Fungsi objektif model += lpSum(upah_per_jam[i-1] * x[(i, j)] for i in range(1,
    16) for j in range(1, 8))
for j in range(1, 8):
    model += lpSum(x[(i, j)] for i in range(1, 16)) >= 8
for i in range(1, 16):
    model += lpSum(x[(i, j)] for j in range(1, 8)) <= 48
from pulp import LpStatus
# Memecahkan model model.solve()
# Mengecek status solusi print("Status Solusi:", LpStatus[model.status])
# Menampilkan jadwal kerja optimal for i in range(1, 16): for j in range(1, 8):
    print(f"Karyawan {i}, Hari {j}: {x[(i, j)].value()} Jam")
# Total biaya tenaga kerja
total_biaya = sum(upah_per_jam[i-1] * x[(i, j)].value() for i in range(1, 16) for j in range(1, 8)) print("Total
    Biaya Tenaga Kerja:", total_biaya)
```



Gambar 1. Implementasi Python

### 3.3. Hasil dan Analisis

1. Efisiensi Biaya: Fungsi objektif memastikan bahwa total biaya tenaga kerja diminimalkan.
2. Kepatuhan terhadap Kendala: Jadwal yang dihasilkan memenuhi semua kendala operasional, termasuk jumlah minimum tenaga kerja harian dan batas maksimum jam kerja per minggu.
3. Fleksibilitas Model: Model ini dapat dengan mudah dimodifikasi untuk menyesuaikan parameter seperti jumlah karyawan, kebutuhan tenaga kerja harian, atau batas jam kerja.

### 4. KESIMPULAN

Penelitian ini menunjukkan bahwa pendekatan pemrograman linier dapat mengoptimalkan penjadwalan kerja secara efisien dengan mempertimbangkan berbagai kendala operasional. Implementasi menggunakan Python dan pustaka PuLP memungkinkan pengambilan keputusan yang lebih cepat dan akurat. Untuk pengembangan lebih lanjut, penelitian dapat mengintegrasikan elemen ketidakpastian dalam model, seperti fluktuasi permintaan tenaga kerja atau variasi tingkat kehadiran karyawan.