

## **Analisis penerapan Bahasa Pemrograman Python dalam perhitungan Pertidaksamaan Linear**

**Yoga Ferdian Anhar<sup>1</sup>, Revival<sup>2</sup>, Sebastian Fhazri Wasono<sup>3</sup>, Shendy Darma Putra<sup>4</sup>, Ebenheser Atakari<sup>5\*</sup>, Perani Rosyani<sup>6\*</sup>**

<sup>1,2,3,4,5</sup> Teknik Informatika, Universitas Pamulang, Tangerang Selatan, Indonesia

Email : [dosen00837@unpam.ac.id](mailto:dosen00837@unpam.ac.id)

(\*: corresponding author)

**Abstrak** - Pemrograman Linear adalah Teknik optimasi matematis yang digunakan untuk memecahkan sistem persamaan dan ketidaksamaan linear dengan memaksimalkan atau meminimalkan suatu fungsi linear. Python menyediakan beberapa pustakan yang kuat untuk pemrograman linear, seperti SciPy, PuLP, dan OR-Tools. Dalam perhitungan pertidaksamaan linear, bahasa pemrograman Python dapat digunakan dengan berbagai cara. Salah satunya adalah dengan menggunakan library seperti PuLP untuk menyelesaikan masalah pemrograman linier. Selain itu, Python juga dapat digunakan untuk membuat grafik fungsi linier menggunakan library seperti matplotlib. Terdapat pula contoh penerapan Python Pemodelan Pemrograman Linier Menggunakan Python PuLP seorang manajer pemasaran yang ingin mengalokasikan anggaran secara optimal di antara saluran media periklanan alternatif seperti radio, televisi, surat kabar, dan majalah. Permasalahan seperti ini dapat dianggap sebagai permasalahan optimasi. Dalam jurnal ini kelompok kami lebih mengerucut pada pustakan pemrograman linear python yaitu PuLP.

**Kata kunci** : analisis, python, pertidaksamaan linear

*Abstract - Linear Programming is a mathematical optimization technique used to solve systems of linear equations and inequalities by maximizing or minimizing a linear function. Python provides several powerful libraries for linear programming, such as SciPy, PuLP, and OR-Tools. In calculating linear inequalities, the Python programming language can be used in various ways. One way is to use libraries such as PuLP to solve linear programming problems. Apart from that, Python can also be used to graph linear functions using libraries such as matplotlib. There is also an example of applying Python Linear Programming Modeling Using Python PuLP a marketing manager who wants to optimally allocate a budget among alternative advertising media channels such as radio, television, newspapers and magazines. Problems like this can be considered as optimization problems. In this journal, our group focuses more on the Python linear programming library, namely PuLP.*

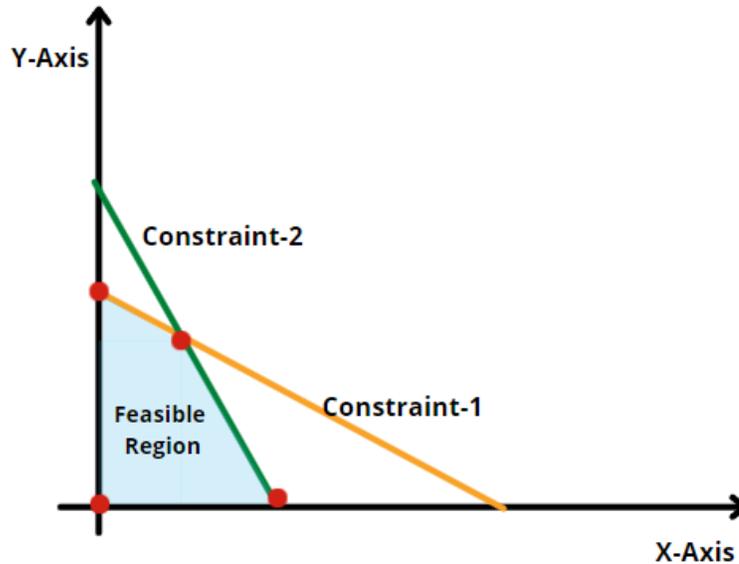
*Keywords: analysis, python, linear inequality*

### **1. PENDAHULUAN**

Dalam beberapa tahun terakhir, banyak perubahan dan peningkatan kebutuhan dan kemampuan teknologi informasi dan komunikasi (ICT) yang diperlukan dalam berbagai bidang, termasuk dalam bidang pendidikan. Dalam konteks pendidikan bahasa pemrograman python populer dan serbaguna.

Jurnal ini akan membahas mengenai penerapan Python Pemodelan Pemrograman Linier Menggunakan Python PuLP seorang manajer pemasaran yang ingin mengalokasikan anggaran secara optimal. Tujuan dari pembahasan ini untuk memecahkan permasalahan optimasi pada kasus tersebut, agar kita dapat mengetahui secara terperinci alur-alur pemecahan masalahnya.

Masalah optimasi dapat direpresentasikan sebagai fungsi matematika yang menangkap trade-off antara keputusan yang perlu dibuat. Solusi yang layak untuk masalah tersebut bergantung pada batasan yang ditentukan dalam bentuk matematika. Pemrograman linier adalah inti dari setiap masalah optimasi. Ini digunakan untuk menyelesaikan berbagai macam perencanaan dan optimalisasi rantai pasokan. Pemrograman linier diperkenalkan oleh George Dantzig pada tahun 1947. Pemrograman ini menggunakan aljabar linier untuk menentukan alokasi optimal sumber daya yang langka.



Pemrograman Linear

## 2. METODE PENELITIAN

Dalam jurnal metode penelitiannya menggunakan Studi Literatur review dan analisa perhitungan tugas mandiri. Metode Studi Literatur review atau Literatur review adalah sebuah metode yang sistematis, eksplisit dan reproduisibel untuk melakukan identifikasi, evaluasi dan sintesis terhadap karya-karya hasil penelitian dan hasil pemikiran yang sudah dihasilkan oleh para peneliti dan praktisi. Kelompok kami menggambil berbagai bahan penelitian dari berbagai sumber seperti google scolar dan website-website lain nya dan juga beberapa buku panduan tentang pertidakamaan dan pemrograman python.

## 3. ANALISA DAN PEMBAHASAN

Pemodelan Pemrograman Linier Menggunakan Python PuLP.

PuLP adalah paket pemodelan Pemrograman Linier dengan tujuan umum dan sumber terbuka dengan Python. PuLP adalah perangkat lunak sumber terbuka gratis yang ditulis dengan Python. Ini digunakan untuk menggambarkan masalah optimasi sebagai model matematika.

PuLP kemudian dapat memanggil salah satu dari banyak pemecah LP eksternal (CBC, GLPK, CPLEX, Gurobi, dll) untuk menyelesaikan model ini dan kemudian menggunakan perintah python untuk memanipulasi dan menampilkan solusinya.

Instal paket pulp:

```
1 | pip install pulp
```

Python

Proses pemodelan PuLP memiliki langkah-langkah sebagai berikut untuk menyelesaikan permasalahan LP:

- Inisialisasi Model
- Tentukan Variabel Keputusan
- Tentukan Fungsi Tujuan
- Tentukan Batasannya
- Pecahkan Model

Pemahaman masalah

Permasalahan ini diambil dari [Pengantar Ilmu Manajemen Oleh Stevenson dan Ozgur](#) .

*Sebuah perusahaan mebel menghasilkan berbagai macam produk. Satu departemen berspesialisasi dalam meja kayu, kursi, dan rak buku. Ini dibuat dengan menggunakan tiga sumber daya tenaga kerja, kayu, dan waktu mesin. Departemen ini memiliki 60 jam tenaga kerja yang tersedia setiap hari, 16 jam waktu mesin, dan 400 kaki papan kayu. Seorang konsultan telah mengembangkan model pemrograman linier untuk departemen tersebut .*

*$x_1$  = jumlah tabel*

*$x_2$  = jumlah kursi*

*$x_3$  = jumlah rak buku*

Fungsi Tujuan: *Untung =  $40 \cdot x_1 + 30 \cdot x_2 + 45 \cdot x_3$*

Batasan:

- *tenaga kerja:  $2 \cdot x_1 + 1 \cdot x_2 + 2,5 \cdot x_3 \leq 60$  Jam*
- *Mesin:  $0,8 \cdot x_1 + 0,6 \cdot x_2 + 1,0 \cdot x_3 \leq 16$  Jam*
- *Kayu:  $30 \cdot x_1 + 20 \cdot x_2 + 30 \cdot x_3 \leq 400$  kaki papan*
- *Tabel:  $x_1 \geq 10$  kaki papan*
- *$x_1, x_2, x_3 \geq 0$*

Inisialisasi Model

Pada langkah ini, kita akan mengimpor semua kelas dan fungsi pulpmodul dan membuat masalah Maximization LP menggunakan kelas LpProblem.

```
Python
1 # Import all classes of PuLP module
2 from pulp import *
3
4 # Create the problem variable to contain the problem data
5 model = LpProblem("FurnitureProblem", LpMaximize)
```

Tentukan Variabel Keputusan

Pada langkah ini, kita akan mendefinisikan variabel keputusan. Dalam soal kita, kita memiliki tiga variabel meja kayu, kursi, dan rak buku. Mari kita buat mereka menggunakan LpVariable kelas. LpVariable akan mengambil empat nilai berikut:

- Pertama, nama arbitrer dari apa yang diwakili oleh variabel ini.
- Kedua adalah batas bawah variabel ini.
- Ketiga adalah batas atas.

- Keempat pada hakikatnya adalah jenis data (diskrit atau kontinu). Pilihan untuk parameter keempat adalah LpContinuous atau LpInteger.

```
Python
1 # Create 3 variables tables, chairs, and bookcases
2
3 x1 = LpVariable("tables", 0, None, LpInteger)
4 x2 = LpVariable("chairs", 0, None, LpInteger)
5 x3 = LpVariable("bookcases", 0, None, LpInteger)
```

#### Tentukan Fungsi Tujuan

Pada langkah ini, kita akan menentukan fungsi tujuan maksimum dengan menambahkannya ke objek LpProblem.

```
Python
1 # Create maximize objective function
2 model += 40 * x1 + 30 * x2 + 45 * x3
```

#### Tentukan Batasannya

Pada langkah ini, kita akan menambahkan 4 batasan yang ditentukan dalam soal dengan menambahkannya ke objek LpProblem.

```
Python
1 # Create three constraints
2 model += 2 * x1 + 1 * x2 + 2.5 * x3 <= 60, "Labour"
3 model += 0.8 * x1 + 0.6 * x2 + 1.0 * x3 <= 16, "Machine"
4 model += 30 * x1 + 20 * x2 + 30 * x3 <= 400, "wood"
5 model += x1 >= 10, "tables"
```

#### Pecahkan Model

Pada langkah ini, kita akan menyelesaikan masalah LP dengan memanggil metode solve(). Kita dapat mencetak nilai akhir dengan menggunakan perulangan for berikut.

```
Python
1 # The problem is solved using PuLP's choice of Solver
2 model.solve()
3
4 # Each of the variables is printed with it's resolved optimum value
5 for v in model.variables():
6     print(v.name, "=", v.varValue)
```

Keluaran:

rak buku = 0,0

kursi = 5,0

tabel = 10,0

#### Studi Kasus: Optimasi Penjadwalan Produksi

#### Deskripsi Masalah:

Sama seperti pada studi kasus sebelumnya, sebuah pabrik memiliki beberapa mesin produksi dan berbagai produk yang harus diproduksi setiap minggu. Setiap produk memerlukan waktu dan sumber daya mesin yang berbeda-beda. Tujuan perusahaan adalah memaksimalkan produksi total mingguan dengan memperhatikan keterbatasan waktu dan kapasitas mesin.

### Model Matematis:

Model matematis sama seperti yang dijelaskan sebelumnya.

### Penerapan Python dengan PuLP:

```
from pulp import LpProblem, LpMaximize, LpVariable, lpSum

# Inisialisasi masalah
model = LpProblem(name="Produksi_Pabrik", sense=LpMaximize)

# Variabel keputusan
x1 = LpVariable(name="x1", lowBound=0, cat="Integer")
x2 = LpVariable(name="x2", lowBound=0, cat="Integer")
x3 = LpVariable(name="x3", lowBound=0, cat="Integer")

# Fungsi objektif
model += 3 * x1 + 5 * x2 + 4 * x3, "Keuntungan_Total"

# Kendala
model += 2 * x1 + x2 + 2 * x3 <= 8, "Mesin_1"
model += x1 + 3 * x2 + 2 * x3 <= 16, "Mesin_2"
model += 4 * x1 + x2 + 2 * x3 <= 12, "Mesin_3"

# Menyelesaikan masalah
model.solve()

# Menampilkan hasil
print("Status Solusi:", LpProblem.status[model.status])
print("Jumlah Produk yang Diproduksi:")
for var in model.variables():
    print(f"{var.name}: {var.value()}")
print("Keuntungan Maksimal:", LpProblem.value[model.objective])
```

### Penjelasan:

Dalam contoh ini, kita menggunakan PuLP untuk mendefinisikan masalah pemrograman linier. Variabel keputusan (`LpVariable`) digunakan untuk menyatakan jumlah produk yang akan diproduksi. Fungsi objektif dan kendala-kendala ditentukan, dan kemudian masalah dipecahkan menggunakan metode `solve()`.

PuLP akan menggunakan solver default (seperti CBC atau GLPK) untuk menemukan solusi optimal dari masalah tersebut.

### Catatan:

Pastikan Anda telah menginstal PuLP sebelum menjalankan kode di atas. Anda dapat menginstalnya dengan menggunakan perintah: `pip install pulp`. Selain itu, Anda dapat memilih solver yang berbeda tergantung pada kebutuhan proyek Anda. Misalnya, untuk menggunakan solver CBC, Anda dapat menambahkan argumen `solver=LpSolverCBC()` saat mendefinisikan masalah.

#### 4. KESIMPULAN

Analisis pada pembahasan ini menerapkan suatu bahasa pemrograman Python dalam memecahkan perhitungan pertidaksamaan linier, Python memberikan perpustakaan yang kuat untuk pemrograman linier seperti halnya Scipy, pulp dan Or-tools. Pustaka pulp dalam Python digunakan untuk memecahkan masalah pemrograman linier dan pengoptimalan. Adapun langkah yang akan terlibat dalam pemodelan pulp ini termasuk menginisialisasi suatu mode, dan mendefinisikan variabel keputusan fungsi yang objektif dan juga pada kendala. Kesimpulan dari analisis penerapan Bahasa Pemrograman Python dalam perhitungan Pertidaksamaan Linear menunjukkan bahwa Python memberikan fleksibilitas dan kemudahan dalam implementasi algoritma matematika kompleks. Penggunaan library PuLP dalam Python dapat memberikan solusi yang efektif untuk perhitungan pertidaksamaan linear. PuLP menyediakan antarmuka yang intuitif untuk pemodelan masalah optimasi, termasuk pertidaksamaan linear. Dengan menggunakan PuLP, kita dapat mendefinisikan variabel, fungsi tujuan, dan batasan-batasan dengan jelas, sehingga mempermudah pemrograman dan analisis hasil optimasi. Overall, penggunaan PuLP dapat meningkatkan keterbacaan dan pemeliharaan kode untuk tugas analisis pertidaksamaan linear menggunakan Python. Bahasa pemrograman python juga dapat digunakan atau diaplikasikan pada perhitungan pertidaksamaan linear ini dan pemrograman python memberikan kemudahan dalam perhitungan pertidaksamaan linear.

#### DAFTAR PUSTAKA

[https://machinelearninggeek-com.translate.goog/solving-linear-programming-using-python-pulp/?x\\_tr\\_sl=en&x\\_tr\\_tl=id&x\\_tr\\_hl=id&x\\_tr\\_pto=tc](https://machinelearninggeek-com.translate.goog/solving-linear-programming-using-python-pulp/?x_tr_sl=en&x_tr_tl=id&x_tr_hl=id&x_tr_pto=tc)

[https://www.researchgate.net/publication/367294575\\_PENERAPAN\\_PEMROGRAMAN\\_PYTHON\\_DALAM\\_MENENTUKAN\\_WAKTU\\_OVERHOUL\\_KONDENSOR\\_TURBIN\\_UAP](https://www.researchgate.net/publication/367294575_PENERAPAN_PEMROGRAMAN_PYTHON_DALAM_MENENTUKAN_WAKTU_OVERHOUL_KONDENSOR_TURBIN_UAP)

<https://air.eng.ui.ac.id/index.php/Cornelius>

<https://www.kompas.com/skola/read/2022/03/16/090000069/pertidaksamaan-linear-satu-variabel-dalam-kehidupan-sehari-hari?page=all>

[https://youtu.be/feXwRk\\_Lezc](https://youtu.be/feXwRk_Lezc)

[Pengantar Ilmu Manajemen Oleh Stevenson dan Ozgur](#)

**Buku PEMROGRAMAN DAN KOMPUTASI NUMERIK MENGGUNAKAN PYTHON**, By Sholihun ,  
Zohan Syah Fatomi

**Buku Pemrograman Metode Numerik Menggunakan Python**, By Windra Swastik.