

**TUGAS MATA KULIAH**  
**OPTIMISASI**  
***NETWORK FLOW PROBLEM USING JULIA***



**Dosen Pengampuh:**  
**Ir. Novalio Daratha, S.T., M.Sc., Ph.D.**  
**Disusun Oleh**  
**Muhammad Choerul Chamdani (G1D021037)**

**PROGRAM STUDI TEKNIK ELEKTRO**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS BENGKULU**  
**2024**

## 1. Network Flow dan Julia

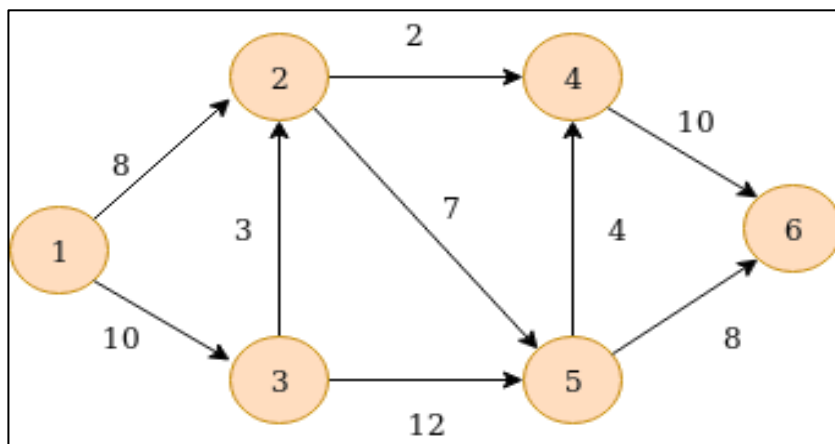
Network Flow atau Aliran Jaringan adalah permasalahan matematika yang terkait dengan menemukan kapasitas maksimal yang dapat dialirkan melalui sebuah jaringan yang terdiri dari simpul dan sisi dengan kapasitas yang bervariasi. Dalam teori grafik, jaringan aliran (juga dikenal sebagai jaringan transportasi) adalah grafik terarah di mana setiap sisi memiliki kapasitas dan setiap sisi menerima aliran. Jumlah aliran pada sisi tidak boleh melebihi kapasitas sisi tersebut. Sering kali dalam penelitian operasi, grafik terarah disebut jaringan, simpulnya disebut simpul dan sisinya disebut busur.

Aliran harus memenuhi batasan bahwa jumlah aliran yang masuk ke simpul sama dengan jumlah aliran yang keluar darinya, kecuali jika itu adalah sumber, yang hanya memiliki aliran keluar, atau tempat penampungan, yang hanya memiliki aliran masuk. Jaringan dapat digunakan untuk memodelkan lalu lintas dalam jaringan komputer, sirkulasi dengan permintaan, cairan dalam pipa, arus dalam rangkaian listrik, atau apa pun yang serupa di mana sesuatu bergerak melalui jaringan simpul.

Julia adalah bahasa pemrograman tingkat tinggi yang dirancang untuk kinerja tinggi, terutama dalam komputasi numerik dan analisis data. Julia memiliki kecepatan yang hampir setara dengan bahasa seperti C dan Fortran, namun tetap mudah dipelajari seperti Python. Bahasa pemrograman Julia juga dapat digunakan untuk menyelesaikan permasalahan network flow problem. Dengan menggunakan library JuMP dan HiGHS yang merupakan library khusus untuk operasi matematika dalam bahasa Julia.

## 2. Contoh Permasalahan Aliran Jaringan

1. Carilah aliran arus maksimal yang dapat mengalir dari simpul 1 menuju ke simpul 6 pada Aliran Jaringan (Network Flow) berikut!



2. Modelkan bentuk Aliran Jaringan di atas ke dalam bentuk matriks 6x6, sebagai berikut.

$A = \begin{bmatrix} 0 & 8 & 10 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 7 & 0 \\ 0 & 3 & 0 & 0 & 12 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10 \\ 0 & 0 & 0 & 4 & 0 & 8 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$

### 3. Menyelesaikan Permasalahan Tersebut dengan Julia

1. Buatlah program Julia untuk menyelesaikannya berdasarkan matriks yang telah dibuat.

```
optimisasijl X network_flows.jl
1 using JuMP
2 using HiGHS
3
4 A = [
5     0 8 10 0 0 0
6     0 0 0 2 7 0
7     0 3 0 0 12 0
8     0 0 0 0 0 10
9     0 0 0 4 0 8
10    0 0 0 0 0 0
11 ] 6x6 Matrix{Int64}:
12
13 n = size(A)[1] 6
14
15 max_flow = Model(HiGHS.Optimizer)  A JuMP Model
16
17 @variable(max_flow, f[1:n, 1:n] >= 0) 6x6 Matrix{VariableRef}:
18
19 @constraint(max_flow, [i = 1:n, j = 1:n], f[i, j] <= A[i, j]) 6x6 Matrix{ConstraintRef{Model, MathOptInterface.ConstraintIndex{MathOptInt
20
21 @constraint(max_flow, [i = 1:n; i != 1 && i != 6], sum(f[i, :]) == sum(f[:, i])) JuMP.Containers.SparseAxisArray{ConstraintRef{Model, Mat
22
23 @objective(max_flow, Max, sum(f[1, :])) f[1,1] + f[1,2] + f[1,3] + f[1,4] + f[1,5] + f[1,6]
24
25 optimize!(max_flow)
26
27 @assert is_solved_and_feasible(max_flow)
28
29 objective_value(max_flow) 14.0
30
```

2. Pada baris pertama dan kedua inialisasikan Packages/Library JuMP dan HiGHS yang akan digunakan.
3. Pada baris keempat deskripsikan Aliran jaringan dalam bentuk matriks A dengan ukuran 6x6.
4. Pada baris ke 15 definisikan 'n' sebagai ukuran dari matriks A.
5. Pada baris ke 17 buat model linear program dengan menggunakan Packages HiGHS.
6. Pada baris ke 19 buat variable 'f' yang mewakili arus setiap edge/busur.
7. Pada baris ke 21 diberi batasan (constraint) bahwa arus yang melalui setiap edge tidak boleh melebihi dari kapasitasnya.
8. Pada baris ke 23 diberi Batasan (constraint) bahwa jumlah arus yang masuk ke setiap simpul (node) harus sama dengan jumlah arus yang keluar dari simpul tersebut.
9. Pada baris ke 25 diberikan tujuan objektif yaitu menemukan arus maksimum yang dapat dialirkan dari simpul sumber ke simpul tujuan.
10. Pada baris ke 27 model yang telah dibuat dan diberikan Batasan – Batasan dioptimalkan dengan perintah 'optimize'.
11. Pada baris ke 29 dilakukan pengecekan status Solusi untuk memastikan bahwa model telah terpecahkan dan memenuhi Batasan yang telah ditetapkan.
12. Pada baris ke 31 tampilkan nilai objektif akhir atau nilai Arus maksimum yang bisa dilalui dengan perintah 'objective\_value', yaitu 14.

Link Github : <https://github.com/kh4irul-id/Optimization>

Link Youtube : <https://youtu.be/179jjTeOpSQ>