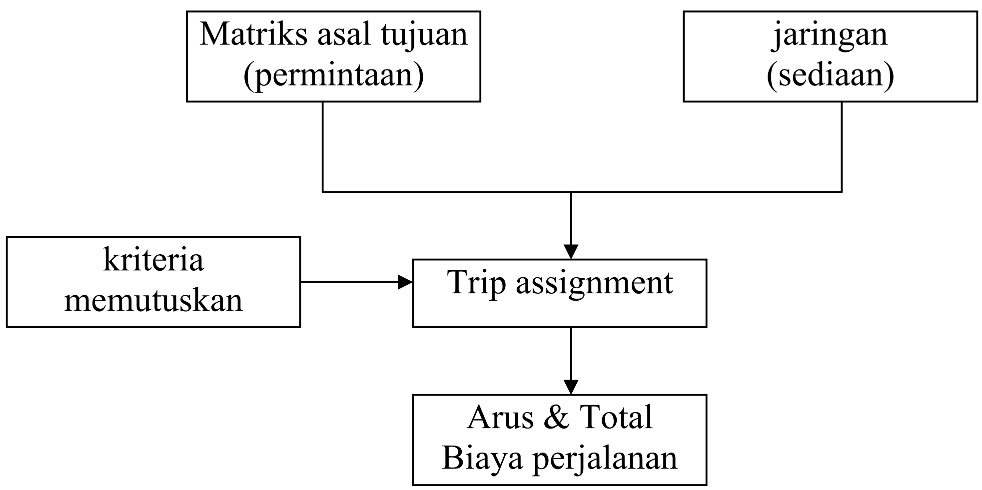
VII. PEMBEBANAN LALU LINTAS
(Trip Assignment)

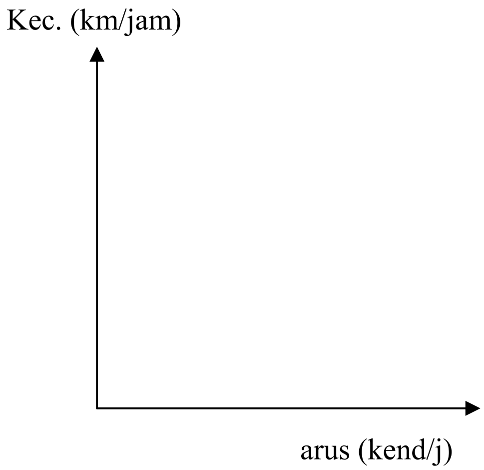
7.1. UMUM

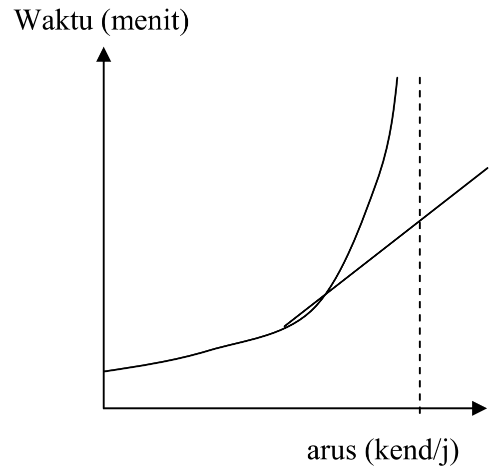
Pembebanan lalulintas (trip assignment) adalah suatu proses dimana permintaan perjalanan (yang didapat dari tahap distribusi) dibebankan ke jaringan jalan. Tujuan trip assignment adalah untuk mendapatkan arus di ruas jalan dan/atau total perjalanan di dalam jaringan yang ditinjau.



7.2. Kurva Kecepatan – arus dan biaya - arus

Hubungan kecepatan-arus sangat sering digunakan dalam rekayasa lalulintas. Konsep ini pada awalnya dikembangkan untuk ruas jalan yang panjang pada jalan bebas hambatan.





Arus lalulintas meningkat → kecepatan cenderung menurun secara perlahan.
Arus mendekati kapasitas → penurunan kecepatan semakin besar.

Model pembebanan rute yang mempertimbangkan kemacetan memerlukan beberapa persamaan (fungsi) yang cocok untuk mengubungkan atribut suatu ruas jalan seperti kapasitas dan kecepatan arus bebas serta arus lalulintas dengan kecepatan dan biaya yang dihasilkan. Hal dinyatakan dalam rumus berikut:

$$C_l = C_l(\{V\})$$

Biaya pada suatu ruas jalan *l* merupakan fungsi dari semua pergerakan *V* pada jaringan jalan tersebut. Rumus cocok untuk daerah perkotaan yang memiliki interaksi yang erat antara arus di ruas jalan dengan tundaan di ruas jalan yang lain. Namun bila kita mempertimbangkan ruas jalan yang panjang, rumus tersebut dapat disederhanakan menjadi:

$$C_l = C_l(V_l)$$

Biaya pada suatu ruas jalan hanya tergantung dari arus dan ciri ruas itu saja. Rumus ini tidak cocok untuk daerah perkotaan yang macet.

Beberapa kurva hubungan biaya-arus:

a. Smock (1976) → Kajian di Detroit

$$t = t_0 \exp\left(\frac{V}{Q_s}\right)$$

t = waktu tempuh per satuan jarak

t_0 = waktu tempuh per satuan jarak pada kondisi arus bebas

Q_S = kapasitas ruas

b. Overgraad (1967)

$$t = t_0 \alpha^{\beta\left(\frac{V}{Q_p}\right)}$$

Q_S = kapasitas ruas; α,β = parameter

c. Departemen Transportasi Inggris

$$T(v) = \begin{cases} d/S_0 & \text{untuk } V < F_1 \\ d/S(V) = \dfrac{d}{S_0 + SS_{01}F_1 - SS_{01}V} & \text{untuk } F_1 \le V \le F_2 \\ d/S_1 + (V/F_2 - 1)/8 & \text{untuk } V > F_2 \end{cases}$$

dimana: $SS_{01} = \dfrac{S_0 - S_1}{F_1 - F_2}$

S_0 = kecepatan arus bebas

S_1 = kecepatan pada arus kapasitas F_2

F_1 = arus maksimum pada kondisi arus bebas masih bertahan

d. IHCM (Indonesian Highway Capacity Manual)

$$V = F_V x\left(1 - (D/D_J)^{(L-1)}\right)^{1/(1-M)}$$

$$\frac{D_0}{D_J} = \left(\frac{(1-M)}{(L-M)} \right)^{1/(L-1)}$$

F_V = kecepatan arus bebas; D = kepadatan; L,M = konstanta

D_0 = kepadatan pada saat kapasitas tercapai

D_J = kepadatan pada kondisi macet total

7.3. Metode pemilihan rute

Faktor yang dipertimbangkan dalam pemilihan rute adalah:

- Waktu tempuh } faktor utama
- Jarak
- Jumlah persimpangan yang dilalui
- Kemacetan
- Rambu lalulintas
- Kondisi permukaan jalan
- Keselamatan
- Dll.

Model pemilihan rute dapat diklasifikasikan berdasarkan beberapa faktor:

- <u>Perbedaan persepsi</u> pribadi tentang apa yang diartikan dengan <u>biaya perjalanan</u> karena adanya perbedaan kepentingan atau informasi yang tidak jelas dan tidak tepat mengenai kondisi lalu lintas.
- Apakah pengaruh kemacetan di ruas jalan diperhitungkan dalam pemodelan.

Kriteria		***Efek stokastik (kesalahpahaman) disertakan?***	
Efek batasan kapasitas (kemacetan) dipertimbangkan		***Tidak***	***Ya***
	Tidak	Al –or-Nothing	Stokastik murni (Dial, Burrel)
	ya	Wardrop Equilibrium)	Stochastic User Equilibrium

Selain itu Robillard (1975) mengklasifikasikan dua metode:

- **Metode proporsional**;
  - Total arus pada suatu ruas jalan (hasil pembebanan) adalah penjumlahan dari semua arus jika setiap pasangan zona dibebankan secara terpisah.
  - Proporsi hasil pembebanan di rute sebanding dengan naiknya tingkat permintaan.
  - Metode A-o-N dan stokastik termasuk dalam kategori ini.
- **Metode tidak proporsional**
  - Kebalikan dari metode proporsional
  - Metode dengan batasan kapasitas masuk dalam kategori ini

7.3.1 All-or-Nothing

Teknik pembebanan ini mengasumsikan bahwa seseorang akan memilih rute berdasarkan pada rute terpendek (shortest path). Pada teknik pembebanan ini, pengaruh kemacetan tidak diperhitungkan, sehingga seberapapun jumlah arus kendaraan tidak mempengaruhi pemilihan rute. Karena itu, metode ini tidak tepat jika digunakan pada jaringan jalan yang macet.

Teknik yang lazim digunakan untuk penentuan rute terpendek adalah Moore, D'Esopo, dan Dijkstra.

7.3.2 Pembebanan Equilibrium

Asumsi dasar dari pemodelan equilibrium adalah masing-masing pengemudi mencoba untuk meminimumkan ongkos perjalanannya. Bagi pengemudi, ongkos dari semua pilihan yang ada diasumsikan diketahui secara implisit dalam pemodelan. Ongkos disini menunjukkan ongkos untuk penggunaan perjalanan, terkadang ongkos ini untuk menunjukkan *generalised cost*, yakni kombinasi dari waktu tempuh, jarak dan ongkos perjalanan lainnya seperti ongkos parkir, terminal, transit, ongkos operasi, kenyamanan, kemudahan dan lain-lain.

Dalam konteks dengan pemilihan rute, pernyataan yang sama dengan asumsi dasar diatas secara singkat telah dibahas oleh Wardrop (1952). Pada tulisan tersebut diuraikan bahwa terdapat dua perilaku intuitif yang menjelaskan bagaimana lalu-lintas dapat didistribusikan kedalam rute yang dikenal dengan Prinsip Wardrop Equilibrium. Dua prinsip tersebut dinyatakan sebagai berikut:

(1) ”*Under equilibrium condition traffic arranges itself in congested networks in such a way that no individual trip maker can reduce his path cost by switching routes.”*

(2) *“Under social equlibrium condition traffic should be arranged in congested networks in such a way that average (or total) travel is minimised.”*

Dari prinsip Wardrop yang pertama dapat disimpulkan bahwa dalam kondisi equilibrium tidak ada pengguna jalan yang dapat mengubah rutenya untuk mendapatkan biaya perjalanan yang lebih murah, karena semua rute yang tidak digunakan mempunyai biaya perjalanan yang sama atau lebih besar dari pada rute yang dilaluinya sekarang. Sehingga dapat dikatakan sistem tersebut mencapai kondisi seimbang menurut pandangan pengguna. Oleh karena itu prinsip ini disebut *user’s equilibrium*. Sedangkan pada prinsip Wardrop yang kedua menyatakan bahwa dalam kondisi optimum, total biaya sistem yang terjadi adalah minimum. Prinsip ini kemudian dikenal dengan *system optimal*. Keduanya saat ini telah menjadi standar praktis dalam setiap evaluasi perencanaan transportasi yang didasarkan pada metode equilibrium.

Pada umumnya arus yang dihasilkan dari dua prinsip tersebut tidak sama, tetapi dalam prakteknya, lalu lintas mengatur dirinya sendiri mengikuti pendekatan prinsip wardrop yang pertama (*user's equilibrium*).

A. Formulasi Pembebanan Equilibrium

Pembebanan dikatakan memenuhi prinsip Wardrop pertama jika semua rute yang digunakan (untuk setiap pasang O – D) harus mempunyai biaya perjalanan yang lebih kecil (minimum) atau sama dibandingkan dengan rute yang tidak digunakan. Secara matematis prinsip tersebut dapat dinyatakan sebagai :

$$c_{pij} \begin{cases} = c_{ij}^{*} & \text{untuk seluruh } T_{pij}^{*} > 0 \\ \geq c_{ij}^{*} & \text{untuk seluruh } T_{pij}^{*} = 0 \end{cases}$$

dimana $\boldsymbol{c_{ij}^{*}}$ adalah biaya minimum dari i ke j. $\boldsymbol{T_{pij}^{*}}$ adalah arus pada lintasan yang memenuhi prinsip Wardrop pertama dan semua biaya dihitung setelah $\boldsymbol{T_{pij}^{*}}$ dibebani. Dalam hal ini arus pada lintasan a dihasilkan dari rumusan berikut :

$$V_a = \sum_{pij} \delta_{pij}^{a} T_{pij} \tag{1}$$

dimana:

$$\delta_{pij}^{a} = \begin{cases} 1 & \text{jika ruas a berada pada lintasan p dari i ke j} \\ 0 & \text{lainnya} \end{cases}$$

Dan biaya sepanjang lintasan dapat dihitung sebagai berikut:

$$C_{pij} = \sum_a \delta^a_{pij} c_a(V^*_a)$$

dimana $\boldsymbol{V_a^*}$ dihitung berdasarkan persamaan (1).

Beckmann (1956) mengajukan rumusan matematis user equilibrium, beliau telah merumuskan kondisi equilibrium sebagai *equivalent convex programming problem* dan telah terbukti bahwa terdapat solusi yang unique, dalam program matematis, rumusannya dinyatakan sebagai berikut:

$$\text{Min} \quad Z\{T_{pij}\} \equiv \sum_a \int_0^{V_a} c_a(v)dv$$

Batasan :

$$T_{pij} \geq 0$$

$$\sum_{pij} T_{pij} = T_{ij}$$

dimana:

T_{ij} = permintaan perjalanan dari asal i ke tujuan j

T_{pij} = arus dari asal i ke tujuan j yang menggunakan lintasan pij

$c_a(v)$ = kurva biaya - arus pada ruas a

Turunan fungsi objektif z terhadap $\boldsymbol{T_{pij}}$ diuraikan sebagi berikut :

$$\frac{\partial Z}{\partial T_{pij}} = \frac{\partial}{\partial T_{pij}} \sum_a \int_0^{V_a} C_a(v)dv$$

$$= \sum_a \frac{d}{dV_a}\left(\int_0^{V_a} C_a(v)dv \right) \frac{\partial V_a}{\partial T_{pij}}$$

Tetapi dari persamaan (1)

$$\frac{\partial V_a}{\partial T_{pij}} = \delta_{pij}^a$$

Karena $\boldsymbol{V_a}$ hanya tergantung pada $\boldsymbol{T_{pij}}$ bila lintasan (pij) melalui ruas a,

$$\frac{d}{dV_a}\int_0^{V_a} C_a(v)dv = C_a(V_a)$$

oleh karena itu,

$$\frac{\partial Z}{\partial T_{pij}} = \sum_a C_a(V_a)\delta_{pij}^a = c_{pij}$$

Turunan kedua dari fungsi objektif z terhadap $\boldsymbol{T_{pij}}$ adalah :

$$\frac{\partial^2 Z}{\partial T_{pij}^2} = \frac{\partial}{\partial T_{pij}} \sum_a C_a(V_a)\delta_{pij}^a$$

$$= \sum_a \frac{dC_a(V_a)}{dV_a}\frac{\partial V_a}{\partial T_{pij}}\delta_{pij}^a$$

$$= \sum_a \frac{dC_a(V_a)}{dV_a}\delta_{pij}^a \delta_{pij}^a \qquad (2)$$

Persamaan (2) ini mempunyai nilai lebih besar atau sama dengan nol hanya jika fungsi turunan hubungan antara biaya–arus bernilai positif atau nol (*non-decreasing functions*).

Turunan kedua, persamaan (2), menunjukkan bahwa fungsi objektif ***z*** adalah fungsi konvex terhadap $\boldsymbol{T_{pij}}$. Sedangkan turunan pertama, menunjukkan bahwa kelandaian (*slope*) di setiap titik pada suatu permukaan yang berkenaan dengan $\boldsymbol{T_{pij}}$ sama dengan biaya sepanjang lintasan tertentu ***pij.***

Perilaku Wardropian yang kedua atau *system optimal*, dimana total biaya perjalanan adalah minimum, dapat dinyatakan secara matematis sebagai berikut:

$$\text{Min} \quad z\{T_{pij}\} \equiv \sum_a v_a c_a(v) \qquad (3)$$

Fungsi ini dapat juga dinyatakan dalam bentuk sebagai berikut :

$$\text{Min} \quad Z\{\mathrm{T}_{\mathrm{pij}}\} = \sum_a \int_0^{v_a} C_{ma}(v)dv \qquad (4)$$

dimana C_{ma} is ongkos marginal perjalanan sepanjang ruas a yang diperoleh dari rumusan berikut :

$$c_{ma} = \frac{\partial[v_a c_a(v_a)]}{\partial v_a} = c_a(v_a) + v_a \frac{\partial c_a(v_a)}{\partial v_a} \qquad (5)$$

Pada bagian yang kanan, terdapat dua terminologi, yang pertama berkaitan dengan biaya rata-rata pada ruas dan yang kedua berkaitan dengan kontribusi tundaan yang ditimbulkan dari kendaraan lain.

Harus dicatat, formulasi *user equilibrium* dan *system optimal* yang dikemukakan oleh Beckmann (1956) diatas memberikan batasan bahwa c_a adalah fungsi dari V_a saja atau "*separable*". Asumsi tersebut mungkin tidak merepresentasikan situasi sebenarnya pada jaringan jalan dalam kota dimana biaya pada ruas *a* merupakan interaksi antara fungsi arus di arus a dengan ruas lainnya (*Non-separable*). Dafermos (1971) mengusulkan perlunya fungsi biaya *non-separable* untuk memodelkan fenomena ini.

Dengan merepresentasikan fungsi objektif z sebagai integral garis, beliau mengusulkan formulasi masalah minimisasi fungsi tersebut sebagai berikut:

$$\text{Min } Z = \sum_{ij}\sum_{pij}\oint^{Tpij} c_{pij}(\bar{u})d\bar{u} = \sum_{a}\oint_{0}^{\bar{v}} c_a(\bar{u})d\bar{u} \qquad (6)$$

dimana $\oint$ menunjukkan integral garis dan $\bar{V}$ adalah vektor dari arus pada seluruh ruas, dengan batasan :

$$\sum_{pij} T_{pij} = T_{ij} \qquad (7)$$

$$T_{pij} \geq 0 \qquad (8)$$

$$V_a = \sum_{pij} \delta_{pij}^{a} T_{pij} \qquad (9)$$

Model Dafermos tersebut mengasumsikan bahwa $c_a = c_a(\bar{V})$ adalah fungsi dari vektor arus pada seluruh ruas dan matriks Jacobian (***J***) dari fungsi biaya ($\partial c_a/\partial V_b$) adalah simetris dan bernilai positif.

Tetapi pendekatan di atas tidak dapat digunakan jika matriks Jacobian (***J***) dari fungsi biaya ($\partial c_a/\partial V_b$) tidak simetris (*asymmetric cost function*). Pendekatan yang lazim digunakan untuk menyelesaikan permasalahan ini adalah metode diagonalisasi (*diagonalisation method*).

B. Mendekati Solusi Equilibrium : Metode Heuristic

Pendekatan dengan metode heuristic ini telah banyak digunakan sebelum pengembangan algoritma penyelesaian program *user equilibrium*. Teknik heuristic dapat digunakan pada jaringan yang kompleks dimana fungsi ongkosnya sangat tergantung pada interaksi arus di ruas tetapi hasil pembebanan tidak dijamin konvergen. Inti prosedur pendekatan ini terletak pada mekanisme pembebanan jaringan jalan. Pada bagian berikut akan dijelaskan tiga metode heuristic yaitu pembebanan dengan penambahan (*incremental assignment*), pembebanan berulang (*iterative assignmnet*) dan pembebanan quantal.

B.1. Pembebanan dengan Penambahan (*Incremental Assignment*)

Pendekatan ini berusaha membagi total matriks perjalanan $\boldsymbol{T}$ menjadi sejumlah bagian matriks dengan menggunakan sekumpulan faktor proporsional $\boldsymbol{p_n}$ sedemikian rupa sehingga $\Sigma_n\ p_n = 1$. Bagian matriks tersebut kemudian dibebani dengan cara penambahan pada pohon (*trees*) yang berturutan. Algoritma penyelesaian prosedur ini dapat diuraikan sebagai berikut :

1. Pilih ongkos awal di ruas, biasanya waktu perjalanan pada *free flow*. Inisialisasi seluruh arus $V_a = 0$; pilih kumpulan fraksi p_n dari matriks perjalanan sedemikian rupa sehingga $\Sigma_n\ p_n = 1$; buat n = 0.
2. Buat kumpulan ongkos minimum *trees* dengan menggunakan ongkos yang terakhir; buat n = n + 1.
3. Bebani $T_n = p_n\ T$ dengan prosedur all-or-nothing; proses ini akan menghasilkan arus F_a ; jumlahkan arus pada setia ruas :
$$V_a^n = V_a^{n-1} + F_a$$
4. Hitung ongkos akhir di ruas berdasarkan arus V_a^n ; jika seluruh fraksi matriks telah terbebani, stop; jika tidak lakukan langkah 2.

Hasil algoritma diatas tidak perlu konvergen terhadap solusi Wardrop's equilibrium. Prosedur pembebanan dengan penambahan ini memiliki keterbatasan dalam hal, jika arus telah dibebani pada suatu ruas, arus tersebut tidak dapat dipindahkan ke ruas yang lainnya; akibatnya jika arus pada permulaan pembenanan terlalu besar, maka hasil algoritma menjadi tidak konvergen.

B.2 Pembebanan Berulang (*Iterative Assignmnet*)

Algoritma pembebanan berulang (*iterative assignmnet*) dikembangkan untuk menyelesaikan masalah penempatan arus yang berlebihan pada kapasitas ruas yang rendah. Algoritma pembebanan ini berusaha menghitung jumlah arus pada suatu ruas dari hasil kombinasi antara arus terakhir pada iterasi sebelumnya dan arus yang dihasilkan dari pembebanan all-or-nothing. Tahapan algoritma ini dapat diuraikan sebagai berikut :

1. Pilih nilai awal ongkos pada ruas, biasanya waktu perjalanan pada saat *free flow*, inisialisasi seluruh arus $V_a = 0$; buat n = 0.
2. Buat pohon (*trees*) ongkos minimum dengan ongkos terakhir, buat n = n + 1.
3. Lakukan pembebanan all-or-nothing seluruh matriks $\boldsymbol{T}$ pada *trees* diatas untuk mendapatkan arus $\boldsymbol{F_a}$.
4. Hitung arus terakhir sebagaimana rumusan berikut :

$$V_a^n = (1 - \phi)\, V_a^{n-1} + \phi F_a$$

5. Hitung ongkos arus terakhir berdasarkan arus V_a^n . Jika arus terakhir tidak mengalami perubahan berarti pada dua iterasi yang berturutan, stop; jika tidak ulangi tahapan 2. Alternative lain dapat digunakan indikator □ untuk menghentikan proses iterasi.

Perbedaan algoritma pembebanan berulang terletak pada pemilihan nilai . Pendekatan yang terbaik untuk memilih nilai ini pernah dilakukan oleh Smock (1962) yaitu dengan menggunakan ϕ = 1/n ; karena itu pendekatan ini dikenal juga sebagai *method of succesive averages* (MSA). Pada metode ini hasil yang dicapai dapat konvergen terhadap Wardrop's equilibrium tetapi diperlukan iterasi yang panjang sehingga dirasakan tidak effisien.

B.3. Pembebanan Quantal

Pada metode konvensional pembebanan matriks O-D pada jaringan jalan adalah dengan menetapkan biaya ruas, menghitung biaya minimum pada lintasan untuk seluruh perjalanan dari asal ke tujuan dan pembebanan perjalananan pada lintasan tersebut. Setiap perubahan biaya di ruas hanya dilakukan pada saat akhir proses ini.

Sebaliknya pada pembebanan quantal dapat mengakomodir perubahan biaya selama prosedur pembebanan. Algoritma pembebanan ini diuraikan sebagai berikut:

1. Buat biaya di ruas pada saat *free flow* dan inisialisasi seluruh $\boldsymbol{V_a} = 0$.
2. Hitung biaya minimum lintasan untuk '**n**' asal perjalanan (*origin*) dan bebani perjalanan $\boldsymbol{T_{ij}}$ pada lintasan ini, perbaharui volume terakhir $\boldsymbol{V_a}$.
3. Apabila seluruh asal perjalanan telah dibebani, stop; jika tidak buat biaya di ruas berdasarkan $c_a(V_a)$ dan kembali ke langkah (2).

Keuntungan metode pembebanan ini adalah bila suatu ruas tertentu dibebani terlalu berlebih pada saat awal pembebanan, biayanya akan bertambah sehingga pada iterasi kedua ruas tersebut menerima lalu lintas lebih sedikit.

Pendekatan ini memungkinkan metode ini menghasilkan penyebaran distribusi perjalanan lebih baik. Oleh Karena itu pada prosedur pembebanan equilibrium, metode pembebanan quantal cenderung menghasilkan nilai awal yang lebih baik dibandingkan dengan pembebanan All-or-nothing. Keuntungan lain dengan pendekatan diatas adalah metode ini cenderung mencegah terjadinya rute yang 'aneh' yang dihasilkan dari pembebanan all-or-nothing. Rute yang 'aneh' ini terjadi bila suatu ruas-ruas tertentu dibebani sangat besar pada saat awal pembebanan sehingga menghasilkan biaya sangat tinggi. Kondisi ini mengakibatkan rute tersebut dikeluarkan dari jaringan.

C. Mencari Solusi Equilibrium: Algoritma Frank - Wolfe

Sebagaimana telah dijelaskan sebelumnya, metoda heuristic mungkin menghasilkan penyelesaian equilibrium yang tidak konvergen. Kenyataan ini yang memotivasi untuk mencari suatu pendekatan yang memformulasikan masalah equilibrium sebagai program matematik. Pada perkembangan selanjutnya Frank dan Wolfe (1956) mengusulkan suatu algoritma menyelesaikan permasalahan eqilibrium ini, yang lazim disebut Algoritma Frank - Wolfe (1956) yang merupakan aplikasi metode kombinasi convex. Algoritma ini merupakan perbaikan dari metode heuristic yang telah dijelaskan sebelumnya. Langkah-langkah algoritma FW dapat diuraikan sebagai berikut:

Step 0: *Inisialisasi.* Set n = 0, dan lakukan pembebanan all-or-nothing(a-o-n) berdasarkan pada kondisi free flow, $c_a(v_a=0)$, untuk menghasilkan $\{F_a^o\}$, yakni arus dibebani terhadap ruas a. Set $v_a^o=F_a^o$.

Step 1: *Update*. Set n = n + 1, dan $c_a^n = c_a(v_a^{n-1})$ a.

Step 2: *Pencarian arah*. Lakukan a-o-n berdasarkan c_a^n menghasilkan arus sementara $\{F_a^n\}$.

Step 3: *Mencari garis*. Cari α_n sedemikian rupa sehingga menimumkan fungsi objektif berikut :

$$\min_{0 \le \alpha_n \le 1} \sum \int_0^{v_a^{n-1} + \alpha_n (F_a^n - v_a^{n-1})} c_a(v)\,dv$$

Step 4: *Pindah*. set $v_a^n = v_a^{n-1} + \alpha_n (F_a^n - v_a^{n-1})$ a

Step 5: *Test konvergensi*. Jika dipenuhi, stop, jika tidak ulangi langkah 1

Untuk mengecek konvergensi, Sheffi (1985) mengusulkan suatu kriteria konvergensi yang didasarkan pada perubahan arus pada iterasi yang berturutan. Formulasinya dinyatakan sebagai berikut :

$$\frac{\sqrt{\sum_a (V_a^{n+1} - V_a^n)^2}}{\sum_a V_a^n} \le K' \tag{10}$$

Perbaikan utama algoritma Frank-Wolfe dibandingkan dengan metode heuristic adalah nilai α_n (dalam metode heuristic dilambangkan dengan ϕ) dihitung dengan menggunakan formulasi program matematis sebagai pengganti dari nilai yang tetap. Karena itu algoritma ini menjamin dapat mencapai tingkat konvergensi dengan lebih effisien.

D. Kriteria Konvergensi

Terdapat tiga tipe dasar kriteria konvergensi pada prosedur pembebanan dengan batasan kapasitas, yaitu :

a. Dengan melihat perbedaan antara arus atau biaya di ruas pada iterasi yang berturutan. Dengan perbedaan ini dilihat apakah proses iterasi selanjutnya akan menghasilkan perubahan yang berarti terhadap arus atau biaya tersebut.

b. Dengan mengukur perbedaan antara asumsi hubungan biaya-arus pada saat awal pembebanan dengan hubungan biaya-arus pada saat akhir pembebanan.

c. Menimbang potensi perbaikan yang dihasilkan apabila dilakukan proses iterasi berikutnya.

Berdasarkan tiga tipe dasar tersebut, beberapa penulis mengusulkan kriteria konvergensi pada pembebanan equilibrium seperti yang diuraikan berikut ini.

(a). Van Vliet (1976) mengusulkan suatu fungsi delta (δ) yang dinyatakan dengan rumusan berikut :

$$\delta = \frac{\sum_{pij} T_{pij}(C_{pij} - C_{ij}^{*})}{\sum_{ij} T_{ij} C_{ij}^{*}} \qquad (11)$$

dimana $\boldsymbol{C_{pij} - C_{ij}^{*}}$ adalah biaya berlebih dari perjalanan pada suatu rute tertentu relatif terhadap biaya minimum dari perjalanan i ke j. Biaya ini dihitung setelah iterasi terakhir.

(b). Evan (1976) menunjukkan bahwa pada iterasi ke – n batas lebih rendah (*lower bound*) fungsi objektif, $\boldsymbol{Z^{*}}$, dapat dihitung dengan rumusan berikut :

$$Z^{*(n)} = Z^{(n)} - \sum_a c_a (V_a^{(n)})(V_a^{(n)} - F_a^{(n+1)}) \qquad (12)$$

$\boldsymbol{Z^{*(n)}}$ tidak perlu bertambah pada setiap iterasi, oleh karena itu estimasi terbaik dari nilai ini adalah $Z^*_{\max}$,yaitu nilai maksimum $\boldsymbol{Z^{*(n)}}$ sampai iterasi yang terakhir. Ukuran ketidakpastian pada fungsi objektif ini adalah:

$$\varepsilon^{(n)} = Z^{(n)} - Z^*_{\max} \qquad (13)$$

Van Vliet (1985) mengusulkan suatu ukuran efektifitas dari iterasi ke –n yaitu seberapa besar pengurangan $\boldsymbol{Z}$ relatif terhadap $\boldsymbol{\epsilon}$**,** dengan rumusan berikut:

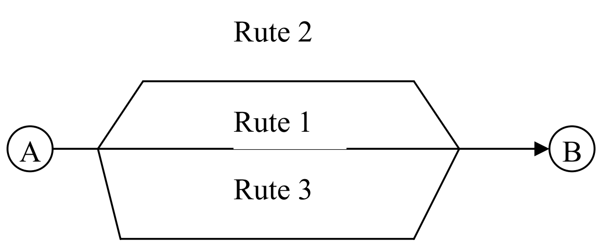
$$F^{(n)} = \frac{Z^{(n)} - Z^{(n-1)}}{\varepsilon^{(n)}} \qquad (14)$$

c). Kriteria konvergensi yang paling praktis didasarkan pada perubahan arus pada iterasi yang berturutan. Kriteria konvergensi ini telah dibahas sebelumnya pada algoritma Frank-Wolfe.

CONTOH PERHITUNGAN PEMBEBANAN PERJALANAN

A. Metode pembebanan dengan penambahan/pembebanan bertahap (incremental assignment).

Terdapat pergerakan sebesar 2000 kendaraan yang akan bergerak dari zona asal A ke zona tujuan B, seperti gambar berikut:



Rute 1 → C= 10 + 0,02 V

Rute 2 → C= 15 + 0,005 V

Rute 3 → C= 12,5 + 0,015 V

Kasus 1:

Bila pergerakan dibagi menjadi empat bagian fraksi dengan persentase seragam 25% (500 kendaraan).

Pembebanan	F	Rute 1		Rute 2		Rute 3	
Ke		Arus	Biaya	Arus	Biaya	Arus	Biaya
0	0	0	***10***	0	15	0	12.5
1	500	500	20	0	15	0	***12.5***
2	500	500	20	0	***15***	500	20
3	500	500	20	500	***17.5***	500	20
4	500	500	***20***	1000	***20***	500	*20*
Total	2000						

Terlihat bahwa hasil pembebanan mencapai kondisi konvergen dengan solusi keseimbangan wardrop. Nilai indikator konvergensi δ adalah:

$$\delta = \frac{\sum_{pij} T_{pij}(C_{pij} - C_{ij}^*)}{\sum_{ij} T_{ij} C_{ij}^*} = \frac{500x(20-20) + 1000x(20-20) + 500x(20-20)}{2000x20} = 0$$

Jika dibandingkan dengan metode pembebanan all-or-nothing yang seluruh pergerakannya (2000 kendaraan) akan menggunakan rute 1, metode pembebanan ini lebih baik.

Kasus 2:

Bila pergerakan dibagi menjadi sepuluh bagian fraksi dengan persentase seragam 10% (200 kendaraan).

Pembebanan Ke	**F**	**Rute 1**		**Rute 2**		**Rute 3**	
		Arus	**Biaya**	**Arus**	**Biaya**	**Arus**	**Biaya**
0	0	0	***10***	0	15	0	12.5
1	200	200	14	0	15	0	***12.5***
2	200	200	***14***	0	***15***	200	15.5
3	200	400	18	0	***15***	200	15.5
4	200	400	18	200	16	200	***15.5***
5	200	400	18	200	***16***	400	18.5
6	200	400	18	400	***17***	400	18.5
7	200	400	***18***	600	***18***	400	18.5
8	200	500	20	700	***18.5***	400	***18.5***
9	200	500	20	800	***19***	500	20
10	200	500	***20***	1000	***20***	500	***20***
Total	2000						

Terlihat bahwa hasil pembebanan mencapai kondisi konvergen dengan solusi keseimbangan wardrop. Nilai indikator konvergensi δ adalah:

$$\delta = \frac{\sum_{pij} T_{pij}(C_{pij} - C_{ij}^{*})}{\sum_{ij} T_{ij} C_{ij}^{*}} = \frac{500x(20-20) + 1000x(20-20) + 500x(20-20)}{2000x20} = 0$$

Kasus 3:

Bila pergerakan dibagi menjadi duapuluh bagian fraksi dengan persentase seragam 5% (100 kendaraan).

Pembebanan ke	**F**	**Rute 1**		**Rute 2**		**Rute 3**	
		Arus	**Biaya**	**Arus**	**Biaya**	**Arus**	**Biaya**
0	0	0	***10***	0	15	0	12.5
1	100	100	***12***	0	15	0	12.5
2	100	200	14	0	15	0	***12.5***
3	100	200	***14***	0	15	100	***14***
4	100	250	***15***	0	***15***	150	14.75
5	100	300	16	50	15.25	150	***14.75***
6	100	300	16	50	***15.25***	250	16.25
7	100	300	16	150	***15.75***	250	16.25
8	100	300	***16***	250	16.25	250	16.25
9	100	400	18	250	***16.25***	250	***16.25***
10	100	400	18	300	***16.5***	300	17
11	100	400	18	400	***17***	300	***17***
12	100	400	18	450	***17.25***	350	17.75
13	100	400	18	550	***17.75***	350	***17.75***
14	100	400	***18***	600	***18***	400	18.5
15	100	450	19	650	***18.25***	400	18.5
16	100	450	19	750	***18.75***	400	***18.5***
17	100	450	19	750	***18.75***	500	20
18	100	450	***19***	850	***19.25***	500	20
19	100	550	21	850	***19.25***	500	20
20	100	550	21	950	***19.75***	500	20
Total	2000						

Terlihat bahwa hasil pembebanan tidak mencapai kondisi konvergen dengan solusi keseimbangan wardrop. Nilai indikator konvergensi δ adalah:

$$\delta = \frac{\sum_{pij} T_{pij}(C_{pij} - C_{ij}^{*})}{\sum_{ij} T_{ij} C_{ij}^{*}} = \frac{550x(21-19{,}75) + 950x(19{,}75-19{,}75) + 500x(20-19{,}75)}{2000x19{,}75} = 0{,}0206$$

Kasus 4:

Bila pergerakan dibagi menjadi empat bagian fraksi dengan persentase tidak seragam (0,4; 0,3; 0,2; 0,1).

Pembebanan	**F**	**Rute 1**		**Rute 2**		**Rute 3**	
ke		**Arus**	**Biaya**	**Arus**	**Biaya**	**Arus**	**Biaya**
0	0	0	***10***	0	15	0	12.5
1	800	800	26	0	15	0	***12.5***
2	600	800	26	0	***15***	600	21.5
3	400	800	26	400	***17***	600	21.5
4	200	800	26	600	***18***	600	*21.5*
Total	2000						

Terlihat bahwa hasil pembebanan tidak mencapai kondisi konvergen dengan solusi keseimbangan wardrop. Nilai indikator konvergensi δ adalah:

$$\delta = \frac{\sum_{pij} T_{pij}(C_{pij} - C_{ij}^{*})}{\sum_{ij} T_{ij} C_{ij}^{*}} = \frac{800x(26-18) + 600x(18-18) + 600x(21{,}5-18)}{2000x18} = 0{,}2361$$

Kasus 5:

Bila pergerakan dibagi menjadi empat bagian fraksi dengan persentase tidak seragam (0,1; 0,2; 0,3; 0,4).

Pembebanan ke	F	Rute 1		Rute 2		Rute 3	
		Arus	Biaya	Arus	Biaya	Arus	Biaya
0	0	0	***10***	0	15	0	12.5
1	200	200	14	0	15	0	***12.5***
2	400	200	***14***	0	15	400	18.5
3	600	800	26	0	***15***	400	18.5
4	800	800	26	800	19	400	***18.5***
Total	2000						

Terlihat bahwa hasil pembebanan tidak mencapai kondisi konvergen dengan solusi keseimbangan wardrop. Nilai indikator konvergensi δ adalah:

$$\delta = \frac{\sum_{pij} T_{pij}(C_{pij} - C_{ij}^*)}{\sum_{ij} T_{ij} C_{ij}^*} = \frac{800x(26-18,5)+800x(19-18,5)+400x(18,5-18,5)}{2000x18,5} = 0,1729$$

Kesimpulan metode pembebanan bertahap:

- ❖ Penggunaan fraksi pentahapan yang semakin kecil secara umum menghasilkan solusi yang mendekati solusi kondisi keseimbangan wardrop. Akan tetapi hal ini tidak selalu benar (seperti kasus 3).
- ❖ Jika salah satu ruas terlanjur mendapat beban yang terlalu besar akibat pembebanan all-or-nothing, maka metode ini akan sangat sulit mengurangi besarnya arus tersebut (seperti pada kasus 4).

B. Pembebanan Berulang (*Iterative Assignment*)

Kasus 1:

Soal sama dengan contoh sebelumnya, gunakan nilai ϕ = 0,5.

Pembebanan ke		ϕ	Rute 1		Rute 2		Rute 3	
			Arus	Biaya	Arus	Biaya	Arus	Biaya
1	Vo		0	***10***	0	15	0	12.5
	F	0.5	2000		0		0	
2	Vo		1000	30	0	15	0	***12.5***
	F	0.5	0		0		2000	
3	Vo		500	20	0	***15***	1000	27.5
	F	0.5	0		2000		0	
4	Vo		250	***15***	1000	20	500	20
	F	0.5	2000		0		0	
5	Vo		1125	32.5	500	17.5	250	***16.25***
	F	0.5	0		0		2000	
6	Vo		562.5	21.25	250	***16.25***	1125	29.38
	F	0.5	0		2000		0	
7	Vo		281.3	***15.63***	1125	20.63	562.5	20.94
	F	0.5	2000		0		0	
8	Vo		1141	32.81	562.5	17.81	281.3	***16.72***
	F	0.5	0		0		2000	
9	Vo		570.3	21.41	281.3	***16.41***	1141	29.61
	F	0.5	0		2000		0	
10	Vo		285.2	***15.7***	1141	20.7	570.3	21.05
	F	0.5	2000		0		0	
	Vo		1143	32.85	570.3	17.85	285.2	***16.78***

Terlihat bahwa hasil pembebanan setelah iterasi ke 10 belum mencapai kondisi solusi keseimbangan wardrop. Hal ini karena kakunya penentuan ϕ.

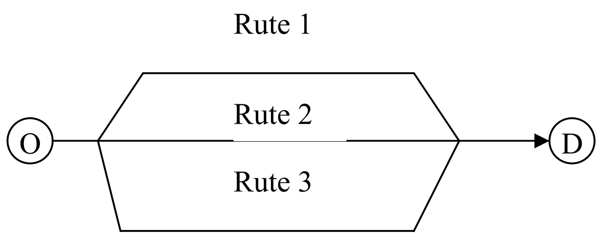
Kasus 2:

Soal sama dengan contoh sebelumnya, gunakan nilai ϕ = 1/n.

Pembebanan ke		ϕ	Rute 1		Rute 2		Rute 3	
			Arus	Biaya	Arus	Biaya	Arus	Biaya
1	Vo		0	***10***	0	15	0	12.5
	F	1	2000		0		0	
2	Vo		2000	50	0	15	0	***12.5***
	F	0.5	0		0		2000	
3	Vo		1000	30	0	***15***	1000	27.5
	F	0.33	0		2000		0	
4	Vo		670	23.4	660	***18.3***	670	22.55
	F	0.25	0		2000		0	
5	Vo		502.5	20.05	995	***19.98***	502.5	20.04
	F	0.2	0		2000		0	
6	Vo		402	***18.04***	1196	20.98	402	18.53
	F	0.167	2000		0		0	
7	Vo		668.9	23.38	996.3	19.98	334.9	***17.52***
	F	0.143	0		0		2000	
8	Vo		573.2	21.46	853.8	***19.27***	573	21.09
	F	0.125	0		2000		0	
	Vo		501.6	20.03	997.1	***19.99***	501.4	20.02

Terlihat bahwa hasil pembebanan setelah iterasi ke 8 hampir mencapai kondisi solusi keseimbangan wardrop.

CONTOH PERHITUNGAN *USER EQUILIBRIUM* (UE) DAN *SYSTEM OPTIMUM* (SO)



Rute 1 → $C_1 = 1 + 3\ V_1$

Rute 2 → $C_2 = 2 + V_2$

Rute 3 → $C_3 = 3 + 2\ V_3$

$T_{OD} = 20$

Hitung arus dan waktu tempuh (biaya) setiap ruas dengan cara UE dan SO.

SOLUSI:

<u>Cara UE</u>

Syarat $\quad C_1 = C_2 = C_3 \quad$ dan $V_1 + V_2 + V_3 = T_{OD} = 20$

$C_1 = C_2 \rightarrow 1 + 3\ V_1 = 2 + V_2 \rightarrow V_1 = 1/3(1 + V_2)$

$C_2 = C_3 \rightarrow 2 + V_2 = 3 + 2\ V_3 \rightarrow V_3 = 1/2(-1 + V_2)$

$V_1 + V_2 + V_3 = 20$

$1/3(1 + V_2) + V_2 + 1/2(-1 + V_2) = 20 \quad \rightarrow V_2 = 11$

$V_1 = 1/3(1 + V_2) = 1/3(1 + 11) \quad \rightarrow V_1 = 4$

$V_3 = 1/2(-1 + V_2) = V_3 = 1/2(-1 + 11) \quad \rightarrow V_3 = 5$

$$\left.\begin{array}{ll} C_1 = 1 + 3\ V_1 = 1 + 3 \times 4 & = 13 \\ C_2 = 2 + V_2 = 2 + 11 & = 13 \\ C_3 = 3 + 2\ V_3 = 3 + 2 \times 5 & = 13 \end{array}\right\} C_1 = C_2 = C_3 \rightarrow \text{OK}$$

<u>Cara SO</u>

Syarat:

$$\min \quad C = \sum_{i=1}^{3} C_i(V_i)V_i \qquad \text{dan } V_1 + V_2 + V_3 = T_{OD} = 20$$

$C = (1 + 3\ V_1)\ V_1 + (2 + \ V_2)\ V_2 + (3 + 2\ V_3)\ V_3$

$V_1 + V_2 + V_3 = T_{OD} = 20 \qquad \rightarrow V_1 = 20 - V_2 - V_3$

$$C = \{1 + 3(20 - V_2 - V_3)\}(20 - V_2 - V_3) + (2 + V_2)V_2 + (3 + 2V_3)V_3$$

$$C = 1180 - 119V_2 - 118V_3 + 6V_2V_3 + 4V_2^2 + 5V_3^2$$

Untuk SO, minimumkan C dengan cara $\partial C / \partial V = 0$

$$\frac{\partial C}{\partial V_2} = 0 \qquad \rightarrow -119 + 6V_3 + 8V_2 = 0 \qquad \text{-------- pers (1)}$$

$$\frac{\partial C}{\partial V3} = 0 \qquad \rightarrow -118 + 6V_2 + 10V_3 = 0 \quad \text{----------- pers (2)}$$

kalikan pers (1) dengan 3 dan pers (2) dengan 4, menghasilkan:

$24V_2 + 18V_3 = 357 \qquad \rightarrow$ pers (3)

$24V_2 + 40V_3 = 472 \qquad \rightarrow$ pers (4)

Kurangi pers (3) dengan (4), menghasilkan

$V_3 = 5{,}23$; dan Subsitusi ke pers (3), maka $V_2 = 10{,}95$

$V_1 \quad = 20 - V_2 - V_3$

$\quad = 20 – 10{,}95 – 5{,}23 = 3{,}82$

Masukkan nilai V tersebut ke C

$C_1 = 1 + 3\ V_1 = 1 + 3 \times 3{,}82 \quad = 12{,}46$

$C_2 = 2 + \ V_2 \ = 2 + \ 10{,}95 \qquad = 12{,}95$

$C_3 = 3 + 2\ V_3 = 3 + 2 \text{ x } 5{,}23 \quad = 13{,}46$

Maka total biaya (waktu) adalah

$$C = \sum_{i=1}^{3} C_i(V_i)V_i = 12{,}46x3{,}82 + 12{,}95x10{,}95 + 13{,}46x5{,}23 = 259{,}796$$

Bandingkan dengan hasil UE:

$$C = \sum_{i=1}^{3} C_i(V_i)V_i = 13x4 + 13x11 + 13x5 = 260{,}000$$

Total biaya pada cara SO lebih kecil dari cara UE → hal ini sesuai dengan harapan.