

PERBANDINGAN KINERJA ALGORITMA *MODIFIED MSA* (*METHOD OF SUCCESSIVE AVERAGE*) UNTUK PEMBEBANAN JARINGAN JALAN

Taufiq Suryo Nugroho
Fakultas Teknik Sipil dan Lingkungan
Institut Teknologi Bandung
taufiq.nugroho@itb.ac.id

Abstract

One of the crucial stages in the four-step model is traffic assignment model which is based on the user equilibrium (UE) principle. Mathematically, the UE assignment can be solved using the convex combination method in which one critical step is the determination of the direction-finding. The method of successive averages (MSA) is the basic method for determining the direction-finding. However, MSA faces long convergence times for large networks and large error for assignments that requires higher accuracy. This paper investigates two modifications of the standard MSA to address these issues, namely the method of successive weighted averages (MSWA) and the self-regulated averaging (SRA) method. These methods were then tested on Sioux Fall network and the road network of Bandung city as a case study respectively. The test results show that the MSWA method can improve the accuracy of the traffic assignment results, while the SRA method can reduce computation time.

Keywords: *four-step model, modified MSA, traffic assignment*

Abstrak

Salah satu tahap penting dalam pemodelan transportasi empat tahap adalah model pembebanan lalu lintas dimana metode yang umum digunakan pada pembebanan lalu lintas adalah model pembebanan kesetimbangan pengguna (*user equilibrium/ UE*). Secara matematis, model pembebanan UE ini dapat diselesaikan dengan metode *convex combination method* dimana metode rata-rata berurut (*MSA / method of successive average*) adalah metode dasar dalam menentukan langkah-pencarian. Namun metode ini terkendala waktu konvergensi yang lama untuk jaringan yang besar dan *error* yang masih tinggi untuk pembebanan dengan tingkat akurasi tinggi. Makalah ini mengembangkan dua modifikasi metode rata-rata berurut untuk mengatasi hal tersebut yaitu metode rata-rata berurut terbebani (*method of successive weighted average / MSWA*) dan metode rata-rata pengaturan sendiri (*self-regulated averaged / SRA*). Kedua metode ini kemudian diuji pada studi kasus jaringan jalan *Sioux Fall* dan jaringan jalan kota Bandung. Hasil pengujian menunjukkan bahwa metode MSWA dapat meningkatkan ketelitian hasil pengujian sedangkan metode SRA dapat menurunkan waktu komputasi yang dibutuhkan untuk mencapai konvergensi.

Kata-kata kunci: modifikasi pembebanan-rerata, pembebanan lalu-lintas, pemodelan empat-langkah

PENDAHULUAN

Pemodelan transportasi empat tahap adalah salah satu pendekatan permodelan transportasi yang paling umum digunakan. Salah satu tahap dari pemodelan transportasi empat tahap adalah tahap pembebanan (*assignment*). Permintaan (*demand*) transportasi yang didapatkan dalam bentuk matriks dibebankan kepada jaringan (*supply*) transportasi (Ortuzar dan Willumsen, 2011). Pembebanan lalu lintas ini umumnya mengikuti hukum Wardrop kedua yang mendefinisikan kondisi keseimbangan pengguna/*user equilibrium* (UE) dalam pembebanan lalu lintas. Kondisi UE ini dinyatakan dalam kondisi yang menunjukkan

seluruh rute yang digunakan oleh pengguna jalan dari satu asal-tujuan memiliki waktu tempuh yang sama atau kurang dari rute-rute lain yang tidak digunakan oleh pengguna (Wardrop, 1952).

Salah satu metode yang paling umum dalam menyelesaikan masalah pembebanan berdasarkan UE adalah dengan menggunakan metode *convex combination* (Sheffi, 1985). Metode *Convex combination* adalah metode yang menggunakan langkah-pencarian (*direction-finding step*) untuk menyelesaikan permasalahan pembebanan UE. Metode ini sangat cocok untuk menyelesaikan permasalahan pembebanan UE karena pencarian langkah dalam *convex combination method* dapat dilakukan secara relatif efisien. Salah satu cara metode dalam penentuan langkah pencarian yang ada di literatur adalah dengan menggunakan *frank-wolfe algorithm* (untuk selanjutnya disebut algoritma FW). Algoritma FW menggunakan parameter langkah-pencarian yang meminimalkan fungsi tujuan dari pembebanan UE (Fukushima, 1984). Dalam metode MSA, alih-alih menggunakan fungsi minimalisasi dalam menemukan langkah-pencarian, langkah-pencarian yang digunakan dalam setiap iterasi ditentukan sebelumnya (Robbins dan Monro, 1951).

Dalam studi terdahulu sudah banyak dilakukan perbandingan antara algoritma FW dengan metode MSA. Sheffi (1985) menyebutkan bahwa algoritma FW akan menemukan solusi pembebanan UE dengan iterasi yang lebih sedikit dibandingkan metode MSA. Hal ini dikarenakan metode algoritma FW akan menemukan nilai pencarian-langkah yang paling efektif untuk menyelesaikan masalah pembebanan UE sementara metode MSA menggunakan langkah-pencarian yang sudah ditetapkan sebelumnya atau dengan kata lain perubahan nilai langkah-pencarian pada tiap iterasi menjadi cenderung statis. Namun dalam hal aplikasi metode algoritma FW lebih sulit diterapkan terutama untuk jaringan yang besar karena memerlukan langkah minimalisasi dalam menentukan nilai langkah-pencarian. Sehingga, sampai saat ini secara praktis dalam jaringan perkotaan atau jaringan dengan ukuran besar metode MSA lebih sering digunakan (Maulana et al., 2023).

Salah satu cara untuk meningkatkan performa metode MSA adalah dengan memodifikasi metode MSA dengan memodifikasi cara menentukan parameter-pencarian langkah untuk setiap iterasi (Botte et al., 2020). Dalam literatur, beberapa cara optimasi metode MSA adalah dengan (1) metode rata-rata berurut terbeban/*method of successive weighted averages* (MSWA) dan (2) metode rata-rata pengaturan sendiri/*self-regulated averaging method* (SRA) (Liu et al., 2009). Dalam studi terdahulu, disebutkan bahwa langkah-pencarian sebaiknya tidak terlalu besar dan tidak terlalu kecil (Szeto et al., 2011). Langkahpencarian yang baik adalah yang memiliki langkah yang memadai sesuai dengan penurunan dan kondisi kurva dari fungsi *convex* pembebanan UE. MSWA mencoba memodifikasi MSA konvensional dengan mengubah langkah-pencarian sesuai dengan beban dalam setiap iterasi. RSA adalah bagian dari metode *adaptive averaging* yang sudah terbukti dapat meningkatkan kecepatan konvergensi dari permasalahan pembebanan UE (Magnanti dan Perakis, 1997). Metode ini lebih cepat mencapai konvergensi karena menggunakan informasi dari iterasi sebelumnya untuk menentukan langkah-pencarian yang lebih optimal

dibandingkan metode MSA konvensional yang menentukan langkah-pencarian dengan cara yang lebih statis.

Tujuan dari makalah ini adalah memodelkan dan mengukur performa kedua model *modified* MSA (MMSA) yang diaplikasikan dalam jaringan sedang dan besar. Beberapa indikator kinerja yang dibandingkan dari hasil pemodelan ini antara lain: jumlah iterasi, waktu komputasi, dan keakuratan hasil pembebanan. Hasil dari pemodelan dan pengukuran ini diharapkan dapat menghasilkan solusi langkah penyelesaian pembebanan lalu lintas yang dapat diimplementasikan dalam jaringan yang besar dengan sumber daya komputasi yang paling optimal. Adapun contoh kasus implementasi jaringan sedang akan menggunakan jaringan Sioux Fall, sedangkan implementasi pada jaringan besar akan menggunakan jaringan kota Bandung. Eksperimen yang dijalankan dalam makalah ini dilakukan didalam program MATLAB 2023.

STUDI PUSTAKA

Permasalahan pembebanan jaringan sesuai dengan prinsip UE menjadi salah satu tahap penting dalam pemodelan transportasi empat tahap. Dalam hal ini, prinsip kesetimbangan Wardrop mulai dapat diimplementasikan sejak Beckmann et al., (1956) menuliskan masalah pembebanan UE dalam bentuk persoalan minimalisasi dalam Persamaan 1.

$$\begin{aligned} \min z(\mathbf{v}) &= \sum_a \int_0^{v_a} t_a(\omega) d\omega \\ \text{s.t } \sum_f p_r^{od} &= q_{od} \quad \forall o, d \\ p_r^{od} &\geq 0 \quad \forall f, o, d \end{aligned} \tag{1}$$

Dengan $z(\mathbf{v})$ adalah fungsi minimalisasi dari pembebanan UE yang merupakan fungsi dari waktu tempuh pada tiap ruas $a(t_a)$ dan arus kendaraan pada ruas $a(v_a)$. Fungsi tujuan tersebut mempunyai dua batasan: (1) Batasan terhadap total permintaan. Jumlah arus kendaraan antara asal-tujuan (q_{od}) haruslah sama dengan penjumlahan total arus dari seluruh rute yang menghubungkan asal-tujuan (2) Batasan positif. Batasan ini menyatakan bahwa arus pada setiap rute r untuk setiap asal tujuan (p_r^{od}) haruslah bernilai lebih besar dari nol/positif.

Sebelum masuk ke dalam penjelasan tentang metodologi penyelesaian pembebanan UE, salah satu langkah yang dibutuhkan dalam pembebanan tersebut adalah pembebanan arus pada rute tercepat. Salah satu algoritma rute tercepat yang paling umum digunakan adalah metode algoritma Dijkstra (Dijkstra, 1959). Dalam metode ini pencarian jalur terpendek dari asal dan tujuan dilakukan dengan menganalisis jarak terpendek dari seluruh rute yang memungkinkan dari asal menuju tujuan. Algoritma Dijkstra ini nantinya digunakan

untuk melakukan pembebanan jalur terpendek atau dikenal dengan pembebanan *All or Nothing* (AoN).

Langkah untuk menyelesaikan persoalan pembebanan UE ini salah satunya adalah dengan mengaplikasikan metode *convex combination* yang salah satunya adalah metode MSA. Secara umum MSA adalah algoritma berbasis iterasi yang menyelesaikan permasalahan pembebanan UE dengan menggunakan arus-bantu yang didapat dari setiap iterasi. Secara umum metode MSA konvensional dapat dilakukan dengan menggunakan langkah-langkah sebagai berikut.

Langkah-0: Inisialisasi. Lakukan pembebanan AoN dengan dasar $t_a = t_a(0)$, $\forall a$, yang akan menghasilkan $\mathbf{v}^1$, buat iterasi $n = 1$, set nilai batas konvergensi $\epsilon = k$

Langkah-1: Pembaharuan. Tetapkan $t_a^n = t_a(v_a^n)$, $\forall a$

Langkah-2: Penetapan arus rute tercepat. Lakukan pembebanan AoN berbasiskan $\mathbf{t}_a^n$ yang akan menghasilkan arus-bantu (*auxiliary flow*) $\mathbf{y}_a^n$

Langkah 3: Penetapan langkah-pencarian. Tetapkan langkah-pencarian α_n sebagai $\alpha_n = \frac{1}{n}$

Langkah 4: Penetapan arus iterasi berikutnya. Tetapkan $v_a^{n+1} = v_a^n + \alpha_n (y_a^n - v_a^n)$, $\forall a$

Langkah 5:es Konvergensi. Jika pada iterasi ke- n konvergensi yang dihasilkan memenuhi, atau dengan kata lain nilai konvergensi ϵ lebih kecil dari nilai batas konvergensi $\epsilon = k$ maka iterasi dihentikan, jika tidak maka kembali ke langkah-0. Nilai epsilon didefinisikan sebagai Persamaan 2.

$$\epsilon = \sum_{od} \frac{|v_{od}^{n+1} - v_{od}^n|}{v_{od}^n} \quad (2)$$

PENGEMBANGAN MODEL MODIFIED MSA

Walaupun algoritma MSA sangat mudah dan praktis untuk digunakan dalam penyelesaian permasalahan pembebanan UE, namun terdapat satu kelemahan MSA yang sangat dirasakan terutama ketika diterapkan pada jaringan yang besar yaitu terkait akurasi dan kecepatan konvergensi algoritma. Hal ini disebabkan karena dua hal: (1) Ukuran langkah-pencarian yang sudah ditentukan sebelumnya terlalu besar sehingga solusi yang dihasilkan pada iterasi berikutnya melampaui nilai optimal yang seharusnya (2) ketika solusi saat ini sudah mendekati solusi optimal, ukuran langkah-pencarian terlalu kecil sehingga menyebabkan proses konvergensi menjadi sangat lambat (Szeto et al., 2011).

Untuk mengatasi hal tersebut, terdapat dua pengembangan model *modified MSA* (MMSA) yang akan dibahas pada makalah ini yaitu (1) metode rata-rata berurut-beban / *method of successive weighted averages* (MSWA) dan (2) metode rata-rata pengaturan sendiri / *self-regulated averaging method* (SRA). Perbedaan dari metode MSA konvensional

dengan metode MMSA ini adalah pada penetapan langkah pencarian. Pada metode MSA konvensional langkah pencarian α_n ditetapkan secara statis mengikuti rumus $\alpha_n = \frac{1}{n}$

Pada metode MSWA dilakukan modifikasi langkah pencarian dengan mempertimbangkan beban sesuai dengan langkah iterasi. Pada makalah ini digunakan metode MSWA dengan Persamaan 3.

$$\alpha_n = \frac{n^k}{1^k + 2^k + 3^k + \dots + j^k}, j = 1, 2, 3, \dots \quad (3)$$

Secara umum Metode MSWA memberikan langkah-pencarian yang lebih teliti dengan memodifikasi nilai alfa pada setiap iterasi. Modifikasi nilai alfa ini dilakukan dengan melakukan pengaturan parameter-beban k yang merupakan *hyperparameter* dari persamaan tersebut. Nilai d menentukan pembebanan yang akan diberikan pada setiap iterasi. Dapat dilihat pada saat k sebagai parameter-beban (*weight parameter*) bernilai 0, maka persamaan tersebut berubah menjadi persamaan MSA konvensional.

Sedangkan pada metode SRA nilai penetapan langkah dilakukan dengan mempertimbangkan hasil pembebanan lalu lintas pada langkah sebelumnya. Hal ini bertujuan agar beban nilai langkah-pencarian α_n yang lebih besar pada periode awal iterasi dibandingkan pada periode akhir iterasi. Hal ini dengan tujuan untuk mempercepat waktu konvergensi di pada periode awal iterasi, dan memperbesar akurasi solusi di periode akhir simulasi. Persamaan berikut mendefinisikan langkah pencarian metode RSA. Pada persamaan ini terdapat parameter Γ dan parameter γ yang mengatur besar nilai β_n yang pada akhirnya sesuai dengan Persamaan 4 akan menentukan besar $\alpha_n = \frac{1}{\beta_n}$.

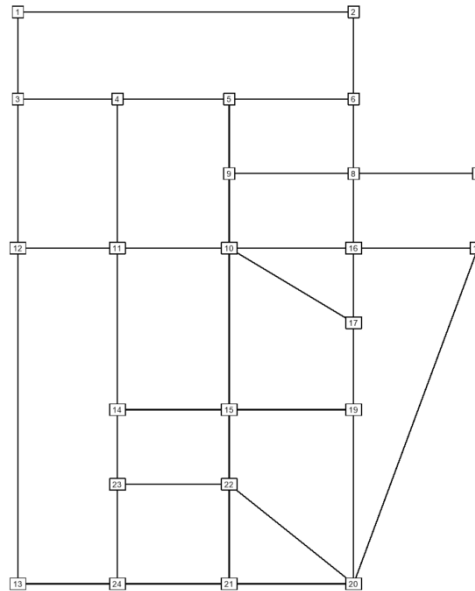
$$\beta_n = \begin{cases} \beta_{n-1} + \Gamma, & \Gamma > 1, \text{ if } \|\mathbf{v}^k - \mathbf{y}^k\| \geq \|\mathbf{v}^{k-1} - \mathbf{y}^{k-1}\| \\ \beta_{n-1} + \gamma, & 0 < \gamma < 1, \text{ if } \|\mathbf{v}^k - \mathbf{y}^k\| < \|\mathbf{v}^{k-1} - \mathbf{y}^{k-1}\| \end{cases} \quad (4)$$

STUDI KASUS

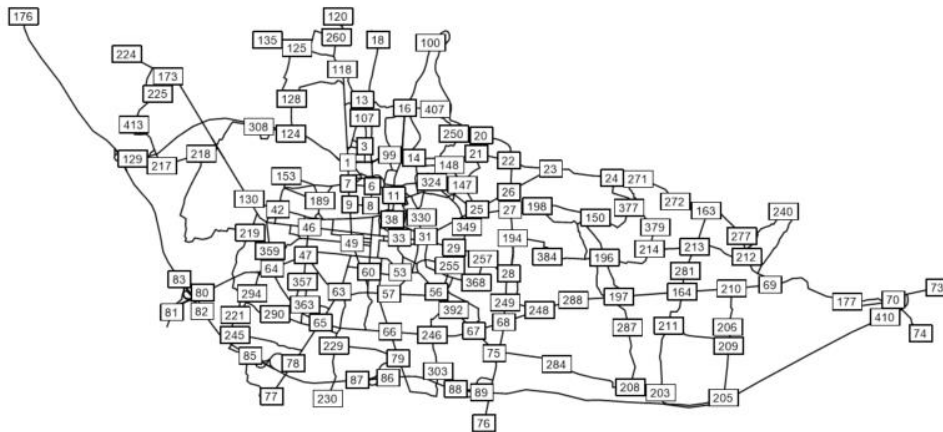
Pada makalah ini studi kasus dibagi menjadi dua tahap (1) studi kasus menggunakan jaringan berukuran sedang (2) Studi kasus menggunakan jaringan berukuran besar. Jaringan berukuran sedang digunakan untuk menentukan pengaturan *hyperparameter* masing-masing model yang akan memberikan performa maksimal. Makalah ini menggunakan jaringan Sioux Fall sebagai studi kasus jaringan sedang. Jaringan Sioux Fall sudah banyak digunakan pada studi tentang jaringan transportasi. Selain berfungsi sebagai studi kasus jaringan sedang, hasil yang dikeluarkan pada jaringan Sioux Fall ini juga akan dijadikan validasi. Proses validasi dilakukan untuk meyakinkan bahwa hasil proses pembebanan yang dilakukan dengan menggunakan metode MSA dan modifikasinya memberikan hasil yang *reliable*. Jaringan Sioux Fall terdiri dari 24 simpul dan 76 ruas serta 576 pasangan asal-

tujuan. Seperti pemodelan jaringan pada umumnya penyiapan jaringan Sioux Fall meliputi penentuan kapasitas, panjang, waktu tempuh pada saat tidak ada arus.

Setelah validasi dan pengaturan *hyper-parameter* dilakukan, metode modifikasi MSA kemudian diterapkan dalam studi kasus jaringan besar. Untuk jaringan besar makalah ini menggunakan jaringan transportasi dunia nyata, yaitu jaringan kota Bandung. Jaringan kota Bandung memiliki 454 simpul dan 1141 ruas serta 22801 pasangan asal tujuan yang dapat menggambarkan kompleksitas jaringan perkotaan pada dunia nyata. Gambar 1 menunjukkan gambaran jaringan Sioux Fall dan jaringan kota Bandung yang digunakan.



(a)



(b)

Gambar 1 (a) Jaringan Sioux Fall dan (b) Kota Bandung

Adapun fungsi tundaan pada studi kasus ini menggunakan fungsi BPR yang sering digunakan dalam studi tentang transportasi. Fungsi BPR dapat dituliskan pada Persamaan 5.

$$t_a = t_0 \left(1 + bpr_{\alpha} \left(\frac{v_a}{cap_a} \right)^{bpr_{\beta}} \right) \quad (5)$$

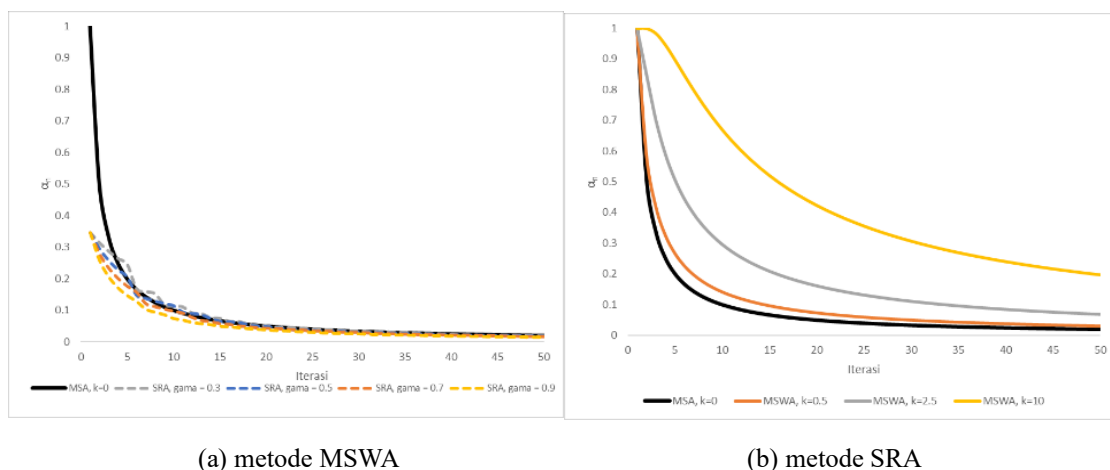
Dapat dilihat bahwa fungsi BPR menghubungkan waktu tempuh akibat adanya volume pada ruas dengan waktu tempuh arus bebas. Selain itu, fungsi BPR mempunyai dua parameter yaitu bpr_{α} dan bpr_{β} yang nilainya bergantung pada karakteristik jalan.

HASIL DAN ANALISIS

Setelah data dikumpulkan, selanjutnya dilakukan analisis pengolahan data. Berikut hasil dan analisis yang dilakukan.

Analisis Pengaturan *Hyperparameter* pada Akurasi dan Performa MMSA pada Jaringan Sedang

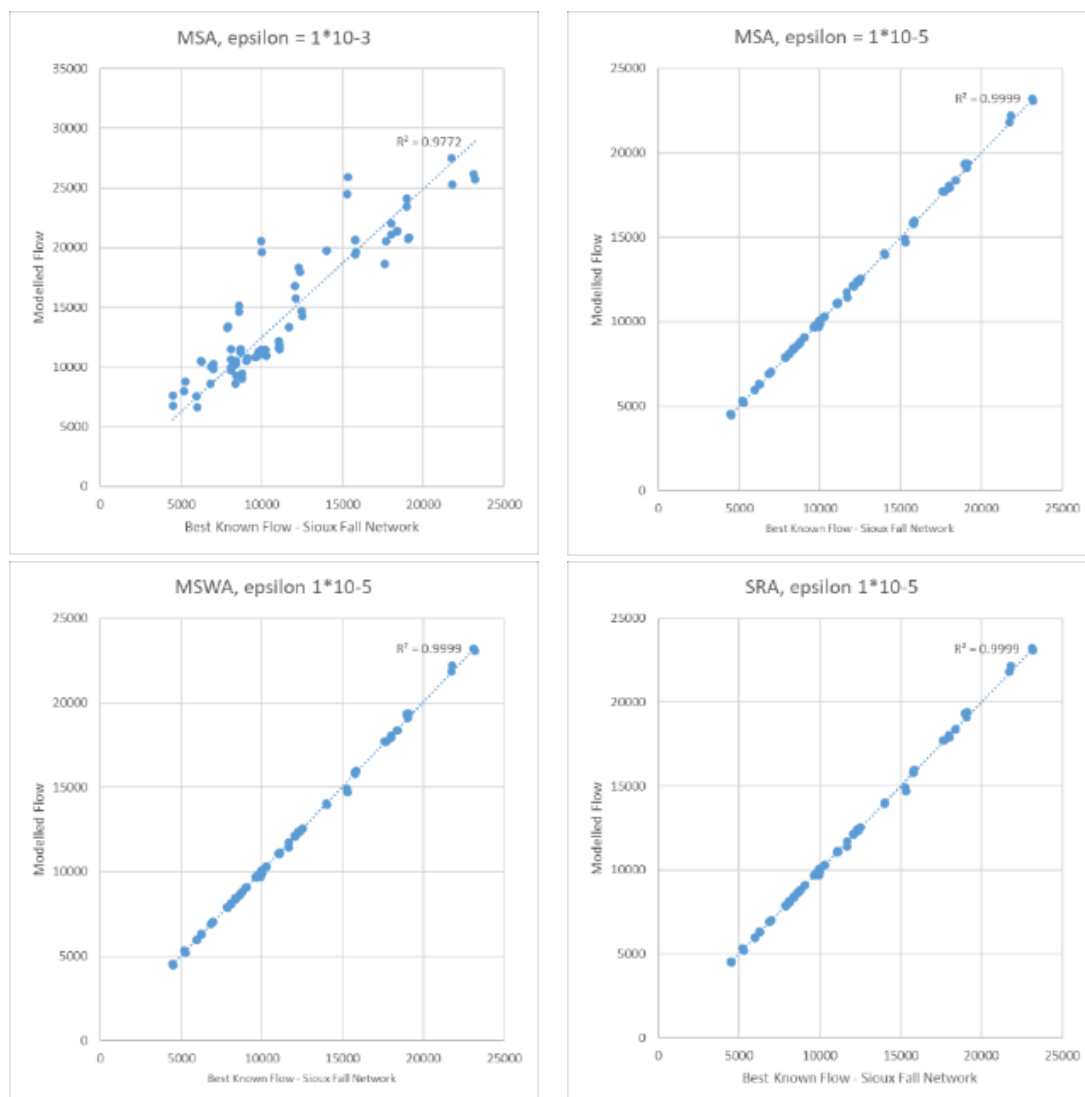
Hasil nilai alfa yang diperoleh dari berbagai macam pengaturan *hyperparameter* baik untuk metode MSA, MSWA, dan RSA dibandingkan. Dalam analisis ini metode MSA dilakukan dengan mengatur nilai *hyperparameter* pada fungsi MSWA dengan nilai 0. Pertama untuk metode MSWA perlu dilakukan analisis sensitivitas dari nilai *hyperparameter* k . Hal ini dilakukan dengan menerapkan variasi nilai k sebagai berikut: 0; 0,01; 0,05; 0,1; 0,5; 1; 2,5; 5; dan 10. Gambar 2 menunjukkan nilai α_n pada lima puluh iterasi pertama sesuai dengan variasi nilai k . Selanjutnya dilakukan pula setiap *hyperparameter* metode RSA yaitu parameter Γ dan γ . Parameter Γ diatur tetap dengan nilai 1.9 sesuai dengan rekomendasi penelitian sebelumnya (Liu et al., 2009). Adapun nilai parameter γ bernilai antara 0 sampai 1. Pada makalah ini dipilih variasi sebagai berikut 0,3; 0,5; 0,7; dan 0,9.



Gambar 2 Nilai Langkah Pencarian pada Berbagai Metode MMSA

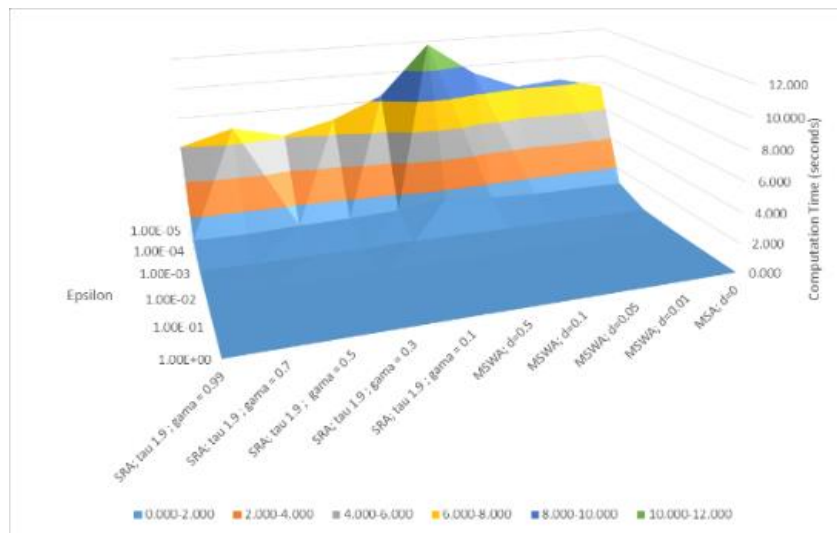
Dari Gambar 2 dapat dilihat bahwa untuk metode MSWA semakin besar nilai k akan menurunkan gradien nilai α_n atau dengan kata lain nilai k yang semakin tinggi akan

meningkatkan jumlah iterasi dan waktu yang diperlukan untuk proses pembebanan UE mencapai nilai konvergensi yang diinginkan. Perhatikan bahwa pada nilai $k=1$ metode MSWA akan menjadi MSA. Sementara itu, untuk metode SRA, dapat dilihat bahwa semakin besar nilai γ , maka nilai awal α_n akan semakin kecil sekaligus menaikkan nilai gradien langkah-pencarian α_n pada awal iterasi. Nilai γ yang besar juga menunjukkan bahwa pada proses akhir iterasi gradien nilai α_n akan mengecil. Hal ini menunjukkan nilai γ yang semakin besar akan semakin sesuai dengan karakteristik nilai langkah-pencarian pada tahap awal iterasi diharapkan perubahan gradien nilai α_n yang besar untuk mempercepat waktu iterasi, namun pada iterasi akhir diharapkan perubahan nilai α_n yang kecil untuk menaikkan akurasi hasil pembebanan UE. Hasil validasi pembebanan UE ditunjukkan pada Gambar 3.

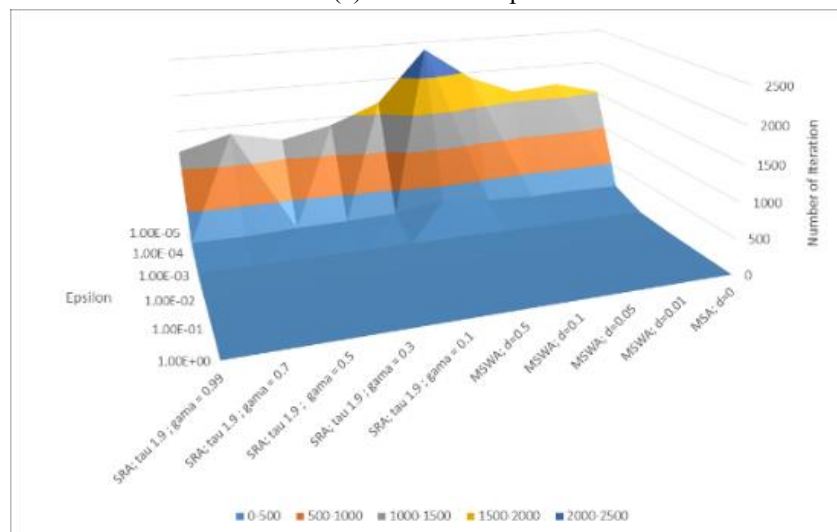


Gambar 3 Validasi Hasil Pembebanan UE pada Beberapa Metode MMSA untuk Beberapa Parameter

Analisis berikutnya adalah analisis perbandingan hasil keluaran pembebanan lalu lintas yaitu volume kendaraan pada ruas (v_a). Perbandingan dilakukan antara hasil pembebanan UE dengan metode MMSA dengan hasil pembebanan UE pada jaringan Sioux Fall yang sudah diketahui/*best-known* yang bersumber dari *Transportation Networks for Research Core Team* (2024). Tujuan dari perbandingan ini adalah untuk mengetahui akurasi dari metode MMSA. Gambar 3 memperlihatkan bahwa pada tingkat epsilon 1×10^{-3} nilai R^2 sudah cukup baik pada angka 0.9771. Namun pada tingkat epsilon 1×10^{-5} nilai R^2 sangat mendekati hasil pembebanan yang diketahui dengan nilai $R^2 = 0,9993$. Hal ini menunjukkan bahwa algoritma yang dibuat sudah dapat menghasilkan keluaran pembebanan yang diharapkan baik itu dengan menggunakan metode MSA, MSWA, maupun SRA. Performa Metode MMSA ditunjukkan pada Gambar 4.



(a) Waktu Komputasi

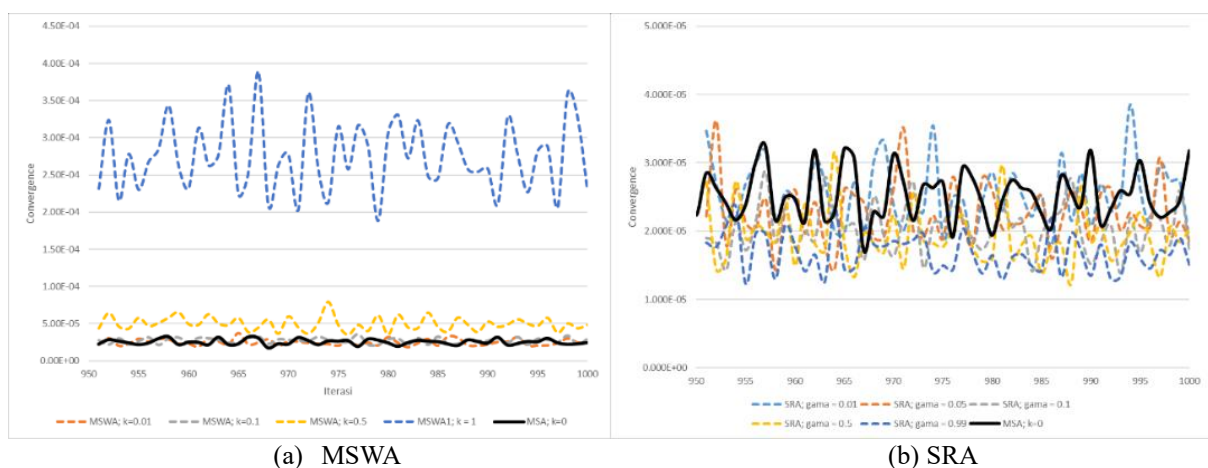


(b) Jumlah Iterasi

Gambar 4 Performa Metode MMSA pada beberapa metode dan pengaturan *hyperparameter*

Analisis selanjutnya pada jaringan sedang Sioux – Fall adalah analisis performa dari pengaturan *hyperparameter* pada berbagai tingkat akurasi/nilai epsilon. Terdapat dua analisis performa yang dilakukan yaitu pertama analisis performa komputasi. Indikator kinerja yang digunakan adalah banyaknya iterasi yang diperlukan dan waktu komputasi yang diperlukan untuk mencapai tingkat akurasi/epsilon tertentu. Gambar 4 menunjukkan jumlah iterasi dan waktu komputasi yang dibutuhkan pada setiap metode akan bergantung pada dengan variasi angka *hyperparameter* yang diterapkan dan tingkat akurasi yang diharapkan. Dibutuhkan nilai *hyperparameter* yang tepat sehingga menghasilkan nilai langkah-pencarian yang mampu menurunkan jumlah iterasi dan waktu komputasi yang dibutuhkan. Hasil analisis juga menunjukkan bahwa pengaturan *hyperparameter* tidak terlalu berpengaruh pada performa algoritma pembebanan dengan nilai ε yang rendah ($\varepsilon \leq 1 \times 10^{-3}$). Namun pada ε yang tinggi ($1 \times 10^{-4} \leq \varepsilon \leq 1 \times 10^{-5}$) pengaturan nilai parameter sangat berpengaruh terhadap performa algoritma. Untuk metode MSWA nilai *hyperparameter* terbaik adalah pada nilai $k = 0,01$, sementara untuk metode RSA nilai *hyperparameter* terbaik adalah pada $\Gamma = 1,9$ dan nilai $\gamma = 0,99$.

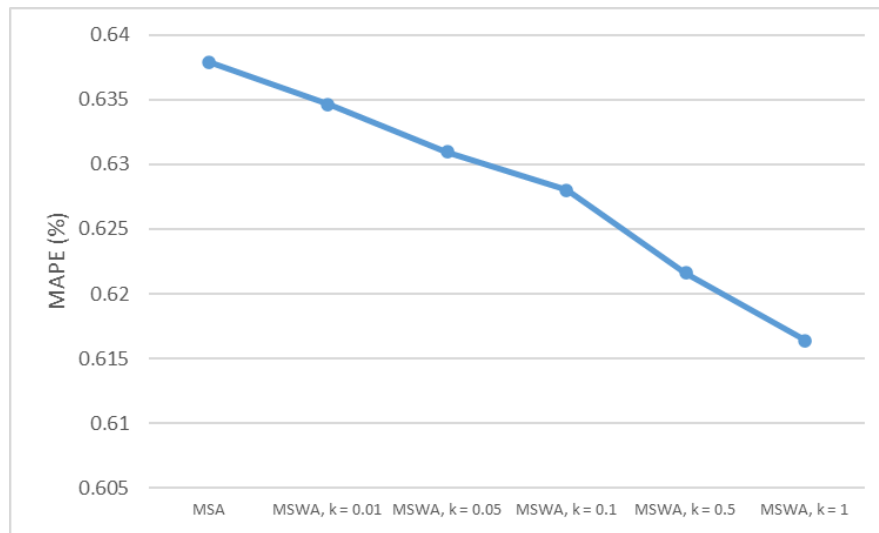
Hasil analisis pada pengaturan nilai *hyperparameter* terbaik menunjukkan performa waktu komputasi yang lebih baik dari MSA konvensional pada metode SRA. Pada metode SRA terdapat penurunan 21% baik dari waktu komputasi dan jumlah iterasi. Namun metode MSWA menunjukkan performa waktu komputasi yang lebih buruk dengan kenaikan sebesar 11 % dari kedua indikator waktu komputasi dan jumlah iterasi. Konvergensi metode MMSA ditunjukkan pada Gambar 5.



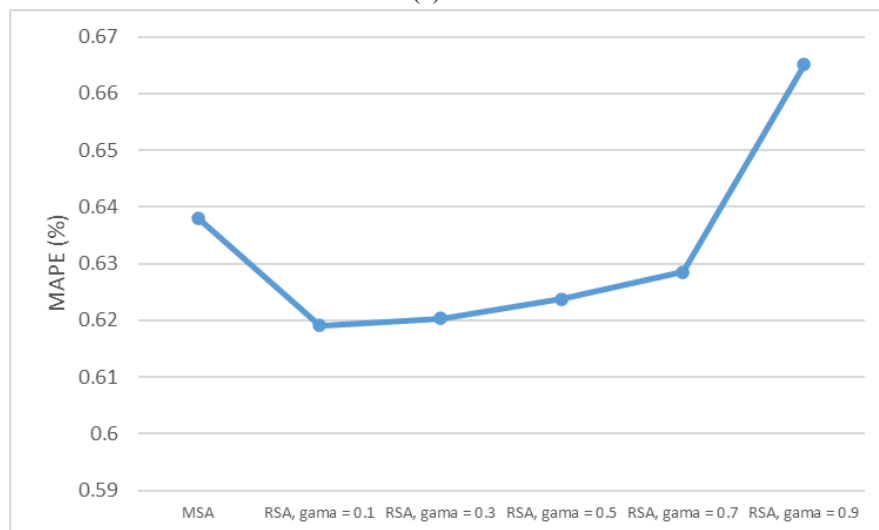
Gambar 5 Konvergensi Metode MMSA pada beberapa metode dan pengaturan *hyperparameter*

Analisis performa yang kedua adalah analisis tingkat akurasi dari metode MSWA dan RSA dengan membandingkan nilai konvergensi dan nilai *error* pada akhir iterasi. Untuk mengamati nilai konvergensi akhir iterasi yang diambil adalah pada iterasi ke-950 hingga iterasi ke-1000. Hasil analisis menunjukkan bahwa pada metode MSWA nilai *hyperparameter* sangat berpengaruh terhadap nilai konvergensi yang dihasilkan. Gambar 5

menunjukkan semakin tinggi nilai *weight parameter* k , maka semakin tinggi *error* yang dihasilkan pada jumlah iterasi yang sama. Hal ini disebabkan nilai k yang besar berimplikasi pada perubahan yang lebih kecil pada nilai langkah pencarian. Hal ini akhirnya menyebabkan kebutuhan iterasi yang jauh lebih banyak untuk mencapai penurunan nilai *error* yang diinginkan. Namun secara teoritis, perubahan nilai langkah pencarian yang kecil mengindikasikan metode MSWA dapat memberikan hasil yang jauh lebih akurat pada jumlah iterasi yang sangat besar, Sementara pada metode RSA nilai *hyperparameter* *gama* tidak terlalu mempengaruhi tingkat akurasi hingga iterasi ke 1000. Perhatikan pula pada metode RSA nilai *error* sudah mendekati nilai *error* yang dihasilkan pada metode MSA konvensional pada akhir iterasi. Akurasi metode MMSA ditunjukkan pada Gambar 6.



(a) MSWA



(b) SRA

Gambar 6 Akurasi Metode MMSA pada beberapa metode dan pengaturan hyperparameter

Sementara untuk nilai *error* digunakan nilai *Mean Absolute Percentage Error* (MAPE) yang mengukur nilai *error* hasil pembebanan terhadap nilai yang sudah diketahui yang dinyatakan dalam persen. Gambar 6 menunjukkan bahwa untuk algoritma MSWA dapat menghasilkan nilai MAPE yang lebih kecil daripada metode MSA, ketika nilai *hyperparameter* k semakin besar. Sementara untuk algoritma SRA, secara umum nilai MAPE yang dihasilkan masih di bawah 1%. Namun perlu diperhatikan bahwa penentuan *hyperparameter* Γ dan γ sangat menentukan akurasi yang dihasilkan pada salah satu percobaan dengan nilai $\gamma = 0,99$ nilai MAPE yang dihasilkan lebih tinggi dari metode MSA.

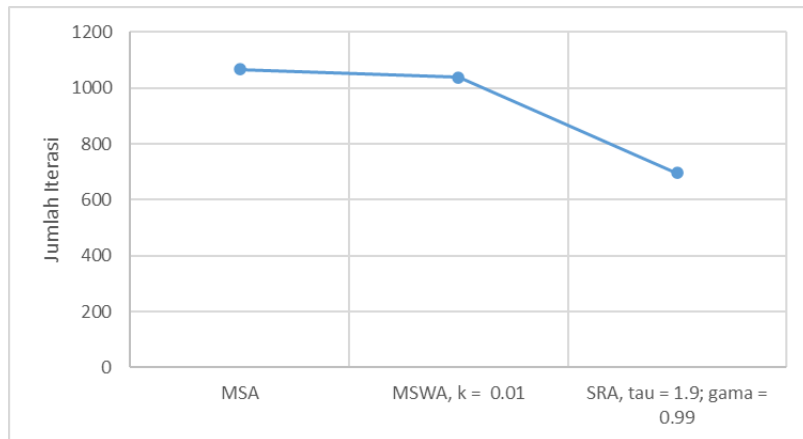
Analisis Hasil pembebanan UE dengan metode MMSA pada Jaringan Besar

Tujuan dari analisis dengan metode MMSA pada jaringan besar adalah membandingkan performa dari metode MSA, MSWA, dan RSA jika digunakan dalam jaringan besar. Perhatikan bahwa untuk percobaan dengan jaringan besar hanya menggunakan pengaturan nilai *hyperparameter* terbaik sesuai dengan Tabel 1. Pengaturan ini dilihat dari sisi waktu komputasi yang diperlukan tanpa mengorbankan akurasi dari hasil pembebanan. Selain itu pembebanan jaringan jalan besar pada kasus ini dilakukan dengan akurasi yang tinggi ($\epsilon > 0,0001$).

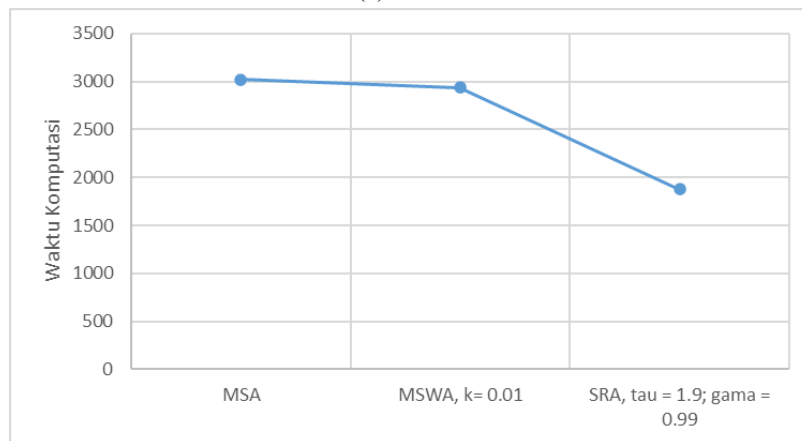
Tabel 1 Nilai *Hyperparameter* Optimal Untuk Metode MMSA

Metode	<i>Hyperparameter</i>
<i>Method of Successive Average</i> (MSWA)	<i>Weight factor</i> , $k = 0,01$
<i>Self-Regulated Average</i> (RSA)	$\tau = 1,9$; $\gamma = 0,99$

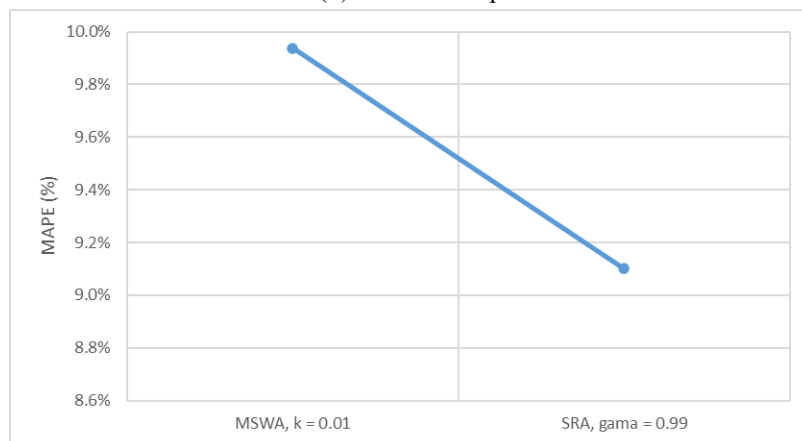
Hasil perbandingan kinerja menunjukkan kesesuaian dengan performa pada percobaan sebelumnya di jaringan sedang, untuk pembebanan dengan nilai epsilon > 0.0001 metode MSWA membutuhkan jumlah iterasi dan waktu komputasi yang cenderung sama dan dibandingkan metode MSA konvensional. Sedangkan untuk metode RSA membutuhkan iterasi yang lebih sedikit dan waktu komputasi yang lebih cepat dibandingkan dengan MSA. Sementara untuk mengukur akurasi, kedua hasil pembebanan algoritma yaitu volume lalu lintas pada ruas dibandingkan dengan hasil dari algoritma MSA. Pengukuran akurasi dilakukan dengan MAPE dan didapatkan nilai MAPE untuk MSWA cenderung lebih besar dibandingkan nilai MAPE dari RSA yaitu 9,9 % dan 9,1 % untuk masing-masing metode. Hal ini mengindikasikan untuk jaringan besar penggunaan metode RSA lebih dianjurkan dibandingkan metode MSWA. Hal ini dikarenakan metode RSA lebih adaptif dalam menentukan nilai langkah pencarian untuk iterasi berikutnya. Gambar 7 menunjukkan hasil perbandingan performa jumlah iterasi, waktu komputasi dan nilai MAPE untuk masing-masing metode.



(a) Jumlah iterasi



(b) Waktu komputasi



(c) Mean of Average Percentage Error (MAPE)

Gambar 7 Perbandingan performa untuk metode MMSA pada jaringan besar

KESIMPULAN

Dari berbagai eksperimen dan percobaan yang telah dilakukan, dapat disimpulkan bahwa modifikasi metode MSA dalam menemukan solusi pembebanan UE dapat

menemukan solusi yang lebih baik. Modifikasi MSA dengan menggunakan pembebanan berurut atau *method of successive weight averaged* (MSWA) secara teoritis dapat menemukan solusi yang lebih akurat baik untuk jaringan sedang dan besar, namun membutuhkan jumlah iterasi yang lebih banyak. Sementara modifikasi dengan mempertimbangkan solusi sebelumnya atau *self-regulated averaged* (SRA) dapat menemukan solusi pembebanan UE dengan jumlah iterasi yang lebih sedikit dan waktu komputasi yang lebih cepat. Penggunaan metode SRA untuk mempercepat proses pembebanan UE ini akan sangat bermanfaat dalam situasi yang memerlukan solusi pembebanan UE dibutuhkan secara berulang seperti penyelesaian *multiuser class assignment problem*, atau persoalan optimalisasi yang membutuhkan kondisi jaringan pada kondisi seimbang seperti *network design problem*.

Walaupun metode MSWA dan metode SRA dapat menemukan solusi yang lebih baik daripada metode MSA, penelitian berikutnya dapat menginvestigasi potensi deret matematika lain sebagai solusi pengganti dari penentuan langkah-pencarian MSA konvensional. Deret tersebut haruslah memenuhi syarat langkah-pencarian yaitu:

$$\sum \alpha_n = \infty \text{ dan } \lim_{n \rightarrow \infty} \alpha_n = 0.$$

Selain itu deret tersebut haruslah deret yang mempunyai gradien yang besar pada awal iterasi untuk mempercepat waktu iterasi, namun memiliki gradien yang semakin kecil seiring jumlah majunya iterasi yang dilakukan. Beberapa deret matematika yang mungkin bisa dipertimbangkan untuk dicoba adalah: deret geometrik, deret pangkat, dan deret peluruhan eksponensial. Deret matematika tersebut juga dapat dikombinasikan dalam pengembangan metode SRA modifikasi sehingga dapat memberikan solusi pembebanan UE yang lebih akurat dengan jumlah iterasi dan waktu komputasi yang lebih singkat.

DAFTAR PUSTAKA

- Beckmann, M., McGuire, C. B., dan B.Winsten, C. 1956. *Studies in the Economics of Transportation*. New Heaven: Yale University Press.
- Botte, M., Gallo, M., Marinelli, M., dan D’Acierno, L. 2020. *A Methodology for Increasing Convergence Speed of Traffic Assignment Algorithms Based on the Use of a Generalised Averaging Function*. MDPI, 10 (16): 1-23.
- Dijkstra, E. W. 1959. A note on Two Problems in Connexion with Graphs. *Numerische Mathematik*, 1: 269–271.
- Fukushima, M. 1984. *A Modified Frank-Wolfe Algorithm for Solving The Traffic Assignment Problem*. Transportation Research Part B: Methodological, 18 (2): 169–177.
- Liu, H. X., He, X., dan He, B. S. 2009. *Method of Successive Weighted Averages (MSWA) and Self-Regulated Averaging Schemes for Solving Stochastic User Equilibrium Problem*. Networks and Spatial Economics, 9 (4): 485–503.

- Magnanti, T. L., dan Perakis, G. 1997. *Averaging Schemes for Variational Inequalities and Systems of Equations*. Mathematics of Operations Research, 22 (3): 568–587.
- Maulana, A., Sjafruddin, A., Frazila, R. B., dan Zukhruf, F. 2023. *Rice Supply Chain Network Equilibrium Optimization Using The Successive Average Method*. Asian Transport Studies, 9: 1-6.
- Ortuzar, J. de D. dan Willumsen, L. G. 2011. Modelling Transport. In *Modelling Transport*. West Sussex: John Wiley & Sons, Ltd.
- Robbins, H. dan Monro, S. 1951. *A Stochastic Approximation Method*. The Annals of Mathematical Statistics, 22 (3): 400–407.
- Sheffi, Y. 1985. *Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Methods*. Transportation Research Part A: General, 20 (1): 76-77.
- Szeto, W. Y., Jiang, Y., dan Sumalee, A. 2011. *A Cell-Based Model for Multi-class Doubly Stochastic Dynamic Traffic Assignment*. Computer-Aided Civil and Infrastructure Engineering, 26 (8): 595–611.
- Transportation Networks for Research Core Team. 2024. *Transportation Networks for Research*. (Online), (<https://Github.Com/Bstabler/TransportationNetworks>, diakses 1 November 2024).
- Wardrop, J. G. 1952. *Road Paper: Some Theoretical Aspects of Road Traffic Research*. Proceedings of the Institution of Civil Engineers, 1 (3): 325–362.

