

BAB II

LANDASAN TEORI

Pada BAB II ini akan disampaikan materi-materi yang berkaitan dengan optimasi *routing* pada jaringan internet dan algoritme *harmony search*, yang merupakan landasan bagi pembahasan penelitian ini.

2.1 Optimasi

Menurut (Budi Santosa, 2017) pengertian dari optimasi bisa dijelaskan sebagai suatu kumpulan formula matematis dan metoda numerik untuk menemukan dan mengidentifikasi kandidat terbaik dari sekumpulan alternatif tanpa harus secara eksplisit menghitung dan mengevaluasi semua alternatif yang mungkin. Menurut (Johanson,2009) berbagai masalah optimasi jaringan muncul di beberapa bidang pekerjaan, termasuk sistem telekomunikasi, transportasi komoditas, perencanaan lalu lintas kereta api,jalan raya dan distribusi tenaga listrik. Optimasi merupakan suatu cara bagaimana mengangkut beberapa entitas secara efisien (paket data, tenaga listrik, kendaraan, dll) dari satu titik ke titik lain dalam jaringan. Masalah optimasi umumnya dapat didekati dengan berbagai cara, misalnya dengan menggunakan pemrograman linier, teori penelitian operasi, simulasi diskrit, dan menggunakan pendekatan algoritmik dari bidang ilmu komputer.

2.1.1 Penyelesaian Masalah Optimasi

Secara umum penyelesaian masalah jalur terpendek dapat dilakukan menggunakan dua metode, yaitu metode konvensional dan metode heuristik. Metode konvensional di hitung dengan perhitungan matematika biasa. Sedangkan metode heuristik dihitung menggunakan sistem pendekatan (Rizal dkk, 2014).

a. Metode Konvensional

Metode konvensional adalah metode yang menggunakan perhitungan matematika biasa, ada beberapa metode konvensional yang bisa digunakan

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

untuk menentukan jalur terpendek, diantaranya algoritme Dijkstra, algoritme *floyd-warshall* dan algoritme *bellman-ford*.

b. Metode Heuristik

Metode heuristik adalah suatu metode yang menggunakan sistem pendekatan dalam melakukan pencarian dalam optimasi. Ada beberapa Algoritme pada metode heuristik yang biasa digunakan dalam permasalahan optimasi, diantaranya algoritme genetika, *ant colony optimization*, logika *fuzzy*, jaringan syaraf tiruan, *tabu search*, *simulated annealing* dan lainlain

2.1.2 Jenis Optimasi

Menurut (Johanson, 2009) Adapun beberapa jenis masalah optimasi yang umum dilakukan dalam jaringan adalah sebagai berikut:

a. *Shortest Path Problem*

Masalah jalur terpendek adalah jalur yang memiliki bobot minimum dari satu simpul ke simpul yang lain dengan tujuan menyelesaikan permasalahan yang berhubungan dengan proses yang membutuhkan optimasi dalam suatu jaringan.

b. *Maximum flow problems*

Maximum flow problems dapat di dekripsikan sebagai masalah pencarian untuk mencari arus maximum yang dapat pada sebuah *network* yang hanya memiliki sebuah *source* dan sebuah *sink*.

c. *Minimum Cost Flow Problem*

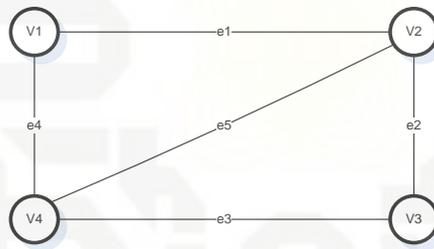
Minimum Cost Flow Problem merupakan usaha untuk meminimumkan suatu jaringan berdasarkan syarat-syarat tertentu. Jaringan akan meminimumkan biaya pendistribusian barang pada suatu jaringan.

2.2 Teori Graf

Graf adalah kumpulan verteks atau node yang dihubungkan satu sama lain melalui sisi/rusuk/busur/*edge*, yang digunakan untuk mempresentasikan objek diskrit dan hubungan antara objek tersebut. Graf G didefinisikan sebagai pasangan himpunan (V,E) , dituliskan dengan $G(V,E)$, penjelasan sebagai berikut :

- a. Dimana V adalah himpunan tidak kosong dari simpul-simpul (titik/verteks/*node*).
- b. Dimana E adalah himpunan sisi (rusuk/*edge*) yang menghubungkan sepasang simpul.

Verteks pada graf dapat berupa objek sembarang seperti kota, atom-atom suatu zat, nama anak, *router*, komponen alat elektronik dan sebagainya. *Edge* dapat menunjukkan hubungan sembarang seperti penerbangan, jalan raya, sambungan telepon, ikatan kimia dan lain-lain(Rizal dkk, 2014). Contoh graf G dapat dilihat pada Gambar 2.1



Gambar 2.1 Graf G dengan 4 simpul dan 5 sisi

Menurut (Rizal dkk, 2014). Didalam teori graf terdapat beberapa istilah yang berkaitan dengan graf antara lain :

- a. Bertetangga (*adjacent*)

Dua buah simpul graf tak berarah G dikatakan bertetangga bila keduanya terhubung langsung oleh sebuah sisi. Dengan kata lain, v_j bertetangga dengan v_k jika (v_j, v_k) adalah sebuah sisi pada graf G .

- b. Lintas (*Path*)

Lintas yang panjangnya n dari simpul awal v_0 ke simpul tujuan v_n di dalam graf G ialah barisan berselang-seling simpul-simpul dan sisi-sisi yang

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

berbentuk $v_0, e_1, v_1, e_2, v_2, \dots, v_{n-1}, e_n, v_n$ sedemikian sehingga $e_1 = (v_0, v_1)$, $e_2 = (v_1, v_2)$, $e_n = (v_{n-1}, v_n)$ adalah sisi-sisi dari graf G . Istilah lain untuk lintasan adalah jalur.

c. Graf Berbobot (*Weighted Graf*)

Graf berbobot adalah graf yang setiap sisinya diberikan bobot. Bobot pada sisi dapat dinyatakan dengan jarak antara dua buah kota, biaya perjalanan antar dua kota, waktu tempuh pesan dari sebuah simpul komunikasi ke simpul komunikasi lain (dalam jaringan komputer), ongkos produksi dan sebagainya.

2.3 Routing

Router adalah salah satu komponen pada jaringan komputer yang mampu melewatkan data melalui sebuah jaringan atau internet menuju sasarannya, melalui proses yang dikenal sebagai *routing*. Proses *routing* dapat dilakukan dengan memasukan informasi suatu alamat secara manual kedalam tabel *routing* ataupun dengan bantuan protokol *routing*. Sebuah *router* mampu mengirimkan data dari satu jaringan ke jaringan lain yang berbeda, *router* hampir sama dengan *bridge*, namun *router* lebih cerdas dibandingkan *bridge*, karena *router* mampu menghubungkan jaringan yang berbeda, sedangkan *bridge* hanya mampu menghubungkan jaringan yang sama. Dalam pengembangan perangkat *router* sejauh ini sudah mencapai bahkan melampaui batas yang diharapkan. *Router* akan mencari jalur terbaik untuk mengirimkan sebuah pesan yang berdasarkan atas alamat tujuan dan alamat asal. *Router* mengetahui alamat secara keseluruhan dari masing-masing komputer dilingkungan jaringan lokalnya, dan *router* lainnya. (Handriyanto, 2009). Beberapa informasi yang dibutuhkan *router* untuk melakukan proses *rotting* :

- a. Alamat tujuan/*destination address* yaitu tujuan atau alamat yang akan di *routing*.
- b. Mengenal sumber informasi yaitu darimana asal informasi yang diperoleh.
- c. Menentukan rute (jalur) yaitu mencari rute mana yang mungkin digunakan untuk mengirimkan informasi sampai ke tujuan.

- d. Pemilihan rute yaitu memilih rute mana yang terbaik untuk diambil agar sampai ketujuan.

2.3.1 Static Routing

Static routing merupakan salah satu pengaturan *routing* yang paling sederhana yang dilakukan pada jaringan komputer. *Static routing* adalah *rute-rute* dari sumber ke jaringan tujuan yang dimasukan secara manual oleh administrator jaringan komputer ke dalam tabel *routing* pada suatu *router*. Yang nantinya bertujuan untuk mem-*forward* paket ke tujuan tertentu. Metode ini memiliki keunggulan dalam menghemat bandwidth jaringan. Penggunaan *static routing* sangat direkomendasikan pada jaringan yang kecil. Namun pada penggunaan jaringan yang berskala besar metode ini sangat tidak efisien karena akan memakan waktu yang lama dalam proses nya. (Edi Doro, 2006)

2.3.2 Dynamic Routing

Dynamic routing adalah suatu protokol *routing* yang mengatur *router-router* sehingga dapat berkomunikasi satu dengan yang lainnya dan saling berbagi informasi untuk dapat mengubah isi *forward table*. Sehingga mampu meneruskan data yang dikirim ke arah yang benar. Routing dinamik yang populer saat ini mengacu pada dua algoritme yang dikenalkan oleh Bellman Ford dengan algoritme *distance vector*nya dan oleh Dijkstra dengan algoritme *link-statenya* (Edi Doro, 2006). Berikut penjelasan kedua algoritme tersebut :

a. Algoritme *Distance Vector*

Algoritme ini bekerja dengan memberikan *router-router* kemampuan untuk mempublikasikan semua jalur yang diketahui keluar ke seluruh interface yang dimilikinya. Algoritme ini menghasilkan dua protokol *routing* yaitu RIP (*Routing Information Protocol*) dan IGRP (*Interior Gateway Routing Protocol*). Namun algoritme ini masih memiliki beberapa kekurangan, salah satunya, lambat dalam memberikan informasi saat terjadinya perubahan rute dari valid ke tidak valid, yang disebut *Route Poisoning* (Edi, 2006).

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

b. Algoritme *Link-State*

Algoritme dasar yang kedua yang digunakan dalam proses *routing* yakni *link-state*. Algoritme *routing* ini dikenal juga sebagai *Shortest Path First* (SPF). Algoritme ini mengelolah database kompleks yang didapat dari informasi topologi. Berbeda dengan *distance vector*, algoritme *link-state* mampu mengelolah secara penuh pengetahuan mengenai jarak *router* dan bagaimana mereka terhubung. Algoritme *link-state* menghasilkan protokol *routing OSPF (Open Shortest Path First)*. Serupa dengan algoritme *distance vector*. *Link-state* juga memiliki kekurangan, karengan algoritme ini bekerja dengan menelusuri setiap rute maka akan memakan memori yang cukup besar (Edi, 2006).

2.4 Topologi *Mesh*

Topologi *mesh* adalah suatu topologi yang memang didisain untuk memiliki tingkat restorasi dengan berbagai alternatif rute atau penjaluran yang biasanya disiapkan dengan dukungan perangkat lunak atau *software*. Komponen utama dalam topologi ini adalah *Digital Cross Connect (DXC)* dengan lebih dari dua sinyal aggregate, dan tingkat cross connect yang beragam. Topologi *mesh* menggunakan prinsip prinsip dari graf lengkap yang mempunyai n titik m dinotasikan K_n . Sehingga jika jumlah verteks pada topologi ini adalah 6 maka jumlah sisinya adalah 15 (Mubarakah, 2007).

2.5 Cisco Packet Tracer

Cisco Packet Tracer adalah program simulasi jaringan yang komplit yang memungkinkan pengguna bereksperimen. *Packet Tracer* menyediakan kemampuan simulasi, visualisasi, authoring, penilaian, dan kolaborasi untuk memfasilitasi pengajaran dan pembelajaran konsep teknologi yang kompleks. *Cisco Packet Tracer* juga merupakan simulator peralatan jaringan *Cisco* yang digunakan sebagai media untuk merancang topologi jaringan komputer.

Cisco Packet Tracer menyediakan dua mode operasi untuk memvisualisasikan perilaku mode real-time jaringan dan mode simulasi. Dalam mode real-time jaringan berperilaku seperti perangkat nyata, dengan respon real-

time langsung untuk semua aktivitas jaringan. Mode real-time memberi alternatif untuk peralatan nyata dan memungkinkan untuk mendapatkan praktik konfigurasi sebelum bekerja dengan peralatan nyata. Dalam mode simulasi, pengguna dapat melihat dan mengendalikan interval waktu, cara kerja dalam transfer data, dan propagasi data di seluruh jaringan. Ini membantu memahami konsep dasar di balik operasi jaringan. Pemahaman yang kuat tentang fundamental jaringan dapat membantu mempercepat pembelajaran tentang konsep terkait (Guterres, 2014).

2.6 Algoritme Harmony Search

algoritme *Harmony Search* (HS) pertama kali diperkenalkan oleh Zong Woo Geem pada tahun 2001. Ide dasar algoritme ini adalah meniru proses perbaikan harmoni musik yang dilakukan oleh sekelompok paduan musik. Ketika sekelompok paduan musik melakukan perbaikan pada harmoni musik yang dimainkan, maka akan terdapat tiga kemungkinan pilihan, antara lain memainkan harmoni musik yang terkenal sesuai ingatan mereka, memainkan musik dengan harmoni sesuai dengan yang asli dengan sedikit penyesuaian atau membuat harmoni musik yang baru. Memformulasikan ketiga pilihan ini pada proses optimasi secara kuantitatif. Ketiga komponen tersebut diformulasikan menjadi penggunaan *harmony memory*, penyesuaian nada, dan proses pembangkitan secara *random*. (Geem, 2001).

Menurut (Geem, 2001). Pada komponen pertama penggunaan *harmony memory* sangat penting karena bisa menjamin bahwa harmoni yang bagus dapat dipertimbangkan sebagai elemen dari vektor solusi terbaru. Komponen kedua yaitu adalah penyesuaian nada dimana mempunyai beberapa parameter seperti pitch adjusting rate (PAR). Penyesuaian nada musik berarti pengubahan frekuensi nada, hal ini berarti membangkitkan nilai yang sedikit berbeda pada algoritme *harmony search*.

Berikut ini adalah langkah-langkah dari algoritme *Harmony Search* dalam menyelesaikan masalah optimasi dalam kasus kombinatorial menurut (Santosa, 2015) :

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

1. Langkah pertama adalah inisialisasi masalah dan parameter algoritme. Berikut ini adalah contoh masalah optimasi. Pada penelitian ini kasus yang ingin diselesaikan adalah pencarian rute terpendek pada jaringan komputer dengan meminimalisasi bobot pada rute yakni *cost* dan *metric*. Dengan bobot yang digunakan pada jalur berupa *bandwidth* dan *delay*. Maka fungsi tujuan yang digunakan adalah sebagai berikut :

a. fungsi objektif untuk *cost*

$$\min f(x_i) = \frac{10^8}{\text{Bandwidth}} \quad (2.1)$$

b. fungsi objektif untuk *metric*

$$\min f(x_i) = 256 * \left(\frac{10^7}{\text{minBW}} + \frac{\text{Total delay}}{10} \right) \quad (2.2)$$

2. Kemudian pada langkah kedua inisialisasi *harmony memory*. Algoritme *harmony search* terdiri dari *harmony memory size*, *harmony memory considering rate*, PAR dan kriteria pemberhentian. contoh matriks *harmony memory* pada gambar 2.2 :

$$\begin{bmatrix} x_1^1 & x_2^1 & \dots & x_{N-1}^1 & x_N^1 \\ x_1^2 & x_2^2 & \dots & x_{N-1}^2 & x_N^2 \\ \dots & \dots & \dots & \dots & \dots \\ x_1^{HMS-1} & x_2^{HMS-1} & \dots & x_{N-1}^{HMS-1} & x_N^{HMS-1} \\ x_1^{HMS} & x_2^{HMS} & \dots & x_{N-1}^{HMS} & x_N^{HMS} \end{bmatrix}$$

Gambar 2.2 Matriks Harmony Memory

3. Langkah ketiga yaitu membangkitkan vektor solusi yang baru (x'_i). Dalam kasus optimasi kombinatorial, misalkan pencarian jalur pada topologi maka ada beberapa langkah modifikasi terhadap algoritme HS. Dalam problem ini ingin ditemukan solusi berupa urutan *node* yang dikunjungi. Misalkan *node* awal pemberangkatan adalah *node* 1 dan *node* berikutnya yang akan dikunjungi berdasarkan dua aturan berikut (Geem.,2015).

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

- a. Nilai x'_k akan dipilih secara random dari titik yang tersimpan pada HM ($\{x_k^1, x_k^2, \dots, x_k^{HMS}\}$) jika bilangan random yang pertama dibangkitkan adalah bernilai kurang dari atau sama dengan HMCR. Sebaliknya jika bilangan random pertama yang dibangkitkan bernilai lebih dari HMCR maka nilai x'_k akan dipilih secara acak dari X_k . Seperti persamaan (2.3).

$$x'_k = \begin{cases} x'_k \in \{x_k^1, x_k^2, \dots, x_k^{HMS}\} & \text{dengan kemungkinan HMCR} \\ x'_k \in X_k & \text{dengan kemungkinan}(1 - HMCR) \end{cases} \quad (2.3)$$

- b. Nilai x'_k akan disesuaikan dengan menggunakan parameter PAR. Proses penyesuaian yang dilakukan adalah dengan penggantian nilai x'_k yang diperoleh pada proses dengan menggunakan parameter HMCR. Nilai x'_k akan digantikan dengan nilai dari tetangga terdekat pada X_k jika nilai bilangan random kedua yang dibangkitkan adalah bernilai kurang dari atau sama dengan $HMCR \times PAR$. Sebaliknya jika bilangan random kedua yang dibangkitkan bernilai lebih dari $HMCR \times PAR$ maka nilai x'_k tidak berubah. Seperti persamaan (2.4).

$$x'_k = \begin{cases} x'_k(k + m) & \text{dengan kemungkinan HMCR} \times PAR \\ x'_k & \text{dengan kemungkinan HMCR} \times (1 - PAR) \end{cases} \quad (2.4)$$

Ket : k adalah indeks pada elemen X_i
 x'_i adalah variabel keputusan ke i
 X_i adalah variabel ke- k pada elemen X_i
 m adalah indeks tetangga (+1 atau -1)

4. Kemudian langkah keempat meng-*update harmony memory*. Apabila nilai vektor yang baru lebih baik dari pada nilai vektor solusi terjelek dilihat dari sudut pandang nilai fungsi tujuan, maka vektor solusi yang baru akan dimasukan kedalam *harmony memory* dan vektor solusi terjelek akan dikeluarkan. Apabila terjadi sebaliknya maka tidak akan terjadi apapun.
5. Langkah kelima dari algoritme *harmony search* adalah mengecek kriteria pemberhentian. Apabila kriteria pemberhentian telah tercapai maka iterasi dihentikan, apabila belum tercapai maka kembali kelangkah ketiga. Terdapat

beberapa macam kriteria pemberhentian antara lain jumlah iterasi maksimal, tidak terjadinya perubahan solusi setelah beberapa iterasi, dan lain-lain.

2.7 Penelitian Terkait

Berikut ini rangkuman penelitian yang berkaitan dengan optimasi *routing* dan penerapan algoritme *Harmony Search* :

Tabel 2.1 Penelitian Terkait

NO	Peneliti/Tahun	Judul	Metode	Hasil
1	Amir I. Amin, Haitham S. Hamzah, Imane A. Saroit, 2013	<i>Harmony Search Based Algorithm For Dynamic Shortest Path Problem In Mobile Ad Hoc Networks.</i>	<i>Harmony Search</i>	Hasil dari penelitian ini menunjukkan bahwa <i>harmony search</i> mampu menemukan rute terbaik pada jaringan wireless sebelum terjadinya perubahan pada topologi.
2	Budi Santoso, 2015.	Penerapan Algoritme <i>Harmony Search</i> Dalam Penyelesaian <i>Resource-Constrained Project Scheduling Problem.</i>	<i>Harmony Search</i>	Hasil yang didapat lagoritma <i>harmony search</i> lebih baik dilihat dari waktu komputasi dibandingkan dengan <i>particle swarm</i>

Hak Cipta Dilindungi Undang-Undang
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

© Hak cipta milik UIN Suska Riau				<i>optimization</i> (PSO).
3.	Effendi M Qilbaaini, Dinar Arnoldus, Aris Kusumawati, Hidayat Syukron	Optimasi Algoritme Genetika Dalam Menentukan Rute Optimal Topologi Cincin Pada <i>Wide Area Network</i>	Algoritme Genetika	Hasil yang telah diuji pada arsitektur WAN dari PT.XYZ terletak di kota waringin barat dengan ukuran populasi 100 dan tingkat mutasi 0,08, hasil yang optimal diperoleh dari <i>final distance</i> adalah 44.
4.	Irine Puspitasari dan Purwanto, 2015	Algoritme <i>Harmony Search</i> Dalam <i>Optimasi Vehicle Routing Problem With Time Window</i> (VRPTW)	<i>Harmony Search</i>	Dari hasil uji coba menyatakan bahwa algoritme <i>harmony search</i> mampu menemukan rute yang optimal dan juga dilakukan perbandingan dengan algoritme <i>tabu search</i> menunjukkan hasil

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

<p>© Hak cipta milik UIN Suska Riau</p>				<p>yang baik, jika dilihat dari total jarak tempuh algoritme harmony search lebih baik yakni 21,3 km, sementara Tabu Search 24,4 km.</p>
<p>5</p>	<p>Kurniwanto Agus, Setijadi Eko, Heri Mauridhi</p>	<p>Analisa Routing Pada Jaringan Data Multi Jalur Menggunakan Metode <i>Ant Colony Optimization</i> (ACO)</p>	<p><i>Ant Colony Optimization</i> (ACO)</p>	<p>Hasil dari pengujian ini dengan membandingkan dengan algoritme <i>Link-State</i>, menghasilkan dari sisi <i>Throughput</i>, algoritme <i>link-state</i> lebih baik dengan selisih sebesar 2,6% pada skenario jalur normal. Sedangkan pada skenario jalur terputus <i>Antnet</i> lebih baik dengan selisih sebesar 3.5 %.</p>

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

<p>6. Hak cipta milik UIN Suska Riau</p>	<p>R. Firasti, Haghighat, M. Mahdavi, 2008</p>	<p><i>Harmony Search Based Algorithm For Bandwidth-Delay-Constrained Least-Cost Multicast Routing.</i></p>	<p><i>Harmony Search</i></p>	<p>Hasil ujicoba menyatakan bahwa algoritme <i>harmony search</i> yang telah dimodifikasi mampu meminimumkan biaya pada jaringan <i>multicast</i>.</p>
<p>7. State Islamic University of Sultan Syarif Kasim Riau</p>	<p>Stanly Henc Dolfi Loppies, Fransiskus Xaverius Manggau, S.Kom., MT, 2014</p>	<p>Algoritme <i>Kruskal</i> Untuk Menentukan Rute Terpendek Pada Jaringan Komputer.</p>	<p>Algoritme <i>Kruskal</i></p>	<p>Hasil yang didapat adalah algoritme <i>kruskal</i> mampu dan efektif dalam menentukan jalur terpendek dengan menghemat pemakaian <i>bandwidth</i>.</p>

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.