

Strategi Routing dalam Jaringan Komputer

Arief Suharsono / 13510087

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

arief.suharsono@comlabs.itb.ac.id

Abstrak—Makalah Strategi Algoritma ini akan membahas mengenai beberapa teknik/strategi yang diterapkan dalam pengelolaan jaringan komputer. Pada makalah ini akan dibahas penerapan langsung beberapa strategi algoritma dalam sistem jaringan komputer. Dimana, sebenarnya jaringan komputer dipandang sebagai sebuah graf dengan milyaran simpul yang memiliki alamat IP (IP Address) dengan milyaran sisi yang menunjukkan interkoneksi antar alamat tersebut, baik secara wired (menggunakan kabel) atau wireless (tanpa kabel). User dari sebuah alamat IP pasti menginginkan koneksi (keterhubungan) dengan user lain dengan alamat IP yang berbeda. Untuk membangun keterhubungan tersebut, dan meningkatkan efisiensi keterhubungan itu, digunakan beberapa strategi algoritma. Dalam makalah ini akan dibahas beberapa penggunaan strategi tersebut, beserta salah satu contoh praktek langsung sebuah strategi.

Kata Kunci—Strategi, Jaringan, Komputer, IP Address, Routing.

I. INTRODUCTION

A. Tentang Strategi Algoritma

Strategi Algoritma adalah pendekatan umum untuk memecahkan persoalan secara algoritmis yang dapat diterapkan pada bermacam-macam persoalan dari berbagai bidang komputasi. Jenis-jenis persoalan yang banyak ditemui dan membutuhkan strategi algoritma adalah persoalan pengurutan dan persoalan pencarian. Dalam kasus ini, saya membahas teknik routing pada jaringan komputer dimana bidang ini memiliki banyak persoalan pencarian. Pengelolaan routing yang baik harus dapat mencari jalan yang menghubungkan antar titik, dimana ketika ada banyak jalan (route) ditemukan, harus ditentukan route yang memiliki cost terkecil untuk meningkatkan efisiensi beserta route alternatif jika route utama tidak dapat dilalui.

Untuk persoalan dengan instansiasi yang besar, solusi dari persoalan tersebut menjadi lebih sulit ditentukan sehingga perlu sebuah prosedur umum yang berisi langkah-langkah penyelesaian persoalan, yang dinamakan algoritma. Dalam kasus ini, persoalan routing pada jaringan komputer termasuk jenis persoalan dengan instansiasi yang besar. Dimana kini seluruh dunia sudah terhubung dengan jaringan internet yang memiliki milyaran titik yang sudah tentu membutuhkan strategi

routing yang menggunakan algoritma-algoritma khusus untuk menjamin interkoneksi dan efisiensi.

II. JARINGAN KOMPUTER

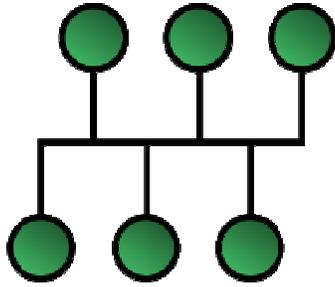
Jaringan komputer adalah koleksi dari komponen hardware dan komputer yang terhubung oleh sebuah jalur komunikasi sehingga antar hardware dan komputer tersebut dapat berbagi informasi dan sumber daya. Jalur komunikasi tersebut dapat berupa media kabel (wired) ataupun nirkabel (wireless).

Jaringan komputer dapat direpresentasikan menjadi sebuah graf berarah maupun graf tidak berarah, dimana setiap simpul menggambarkan sebuah titik, yang dapat berupa komputer atau perangkat keras lainnya (router, server, atau lainnya), dan setiap sisi menggambarkan media interkoneksi antar titik tersebut. Setiap titik harus memiliki media interkoneksi jika ingin berkomunikasi dengan titik lainnya, ibarat sebuah simpul harus memiliki lintasan ke simpul lainnya untuk bisa terhubung dengan simpul tersebut.

Cara penempatan titik dan media interkoneksi tersebut bermacam-macam, bergantung kepada topologi jaringan tersebut. Topologi jaringan adalah gambaran umum dari interkoneksi antar titik (simpul) dari sebuah jaringan komputer. Berdasarkan topologinya, jaringan komputer ini dapat diklasifikasikan menjadi beberapa macam, dimana setiap macam memiliki representasi graf yang berbeda.

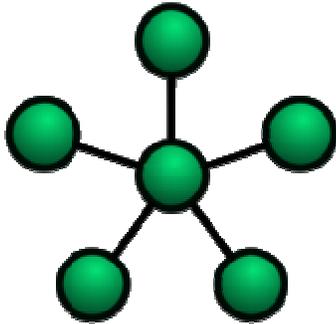
- Topologi Bus.

Dalam topologi ini, semua titik terhubung dalam sebuah medium dan semua titik berkomunikasi melalui medium ini. Dimana medium yang dimaksud tersebut dapat direpresentasikan menjadi sebuah “simpul semu” yang bertetangga dengan semua simpul.



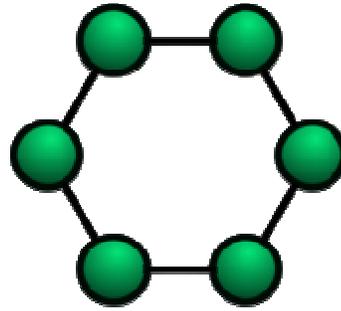
Gambar 2 : Ilustrasi Topologi Bus

- Topologi Star.
 Dalam topologi ini, semua titik terhubung ke sebuah titik spesial yang menjadi pusat jaringan, dimana titik spesial ini berfungsi menjadi pengatur komunikasi dari titik-titik lainnya. Topologi ini dapat ditemukan dalam Wireless LAN, dimana semua komputer (sebagai titik-titik biasa) terhubung ke dalam sebuah Wireless Access Point (sebuah titik spesial yang menjadi pusat).
 Topologi ini banyak digunakan karena jika salah satu titik mati (asalkan bukan titik spesial), tidak mempengaruhi interkoneksi dari titik-titik lainnya.



Gambar 3 : Ilustrasi Topologi Star

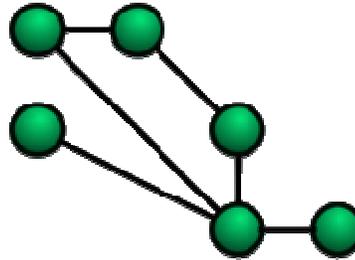
- Topologi Ring.
 Dalam topologi ini, semua titik terhubung ke titik yang ada di kiri dan kanannya, sehingga setiap titik akan dapat menjangkau titik lainnya dengan menyampaikan data ke titik sebelahnya, untuk disampaikan ke titik sebelahnya lagi, dan terus berulang hingga data sampai ke titik tujuan. Topologi ini digunakan dalam Fiber Distributed Data Interface (FDDI). Topologi ini dapat direpresentasikan sebagai sebuah graf lingkaran. Topologi ini dapat juga dikatakan sebagai graf euler dan graf hamilton.



Gambar 4 : Ilustrasi Topologi Ring

Topologi Ring ini memiliki kelemahan, dimana jika salah satu titik mati, jaringan menjadi cacat, dimana ada sebuah titik yang tidak dapat berkomunikasi dengan titik-titik tertentu. Hal ini diibaratkan jika sebuah graf melingkar, salah satu simpulnya dihilangkan, graf tersebut menjadi tidak terhubung.

- Topologi Mesh.
 Dalam topologi ini, semua titik terhubung sembarang dengan titik yang lain, dengan syarat sebuah titik harus memiliki lintasan ke setiap titik yang lain. Topologi ini dapat direpresentasikan sebagai sebuah graf terhubung dimana tidak ada satupun simpul yang terpencil.

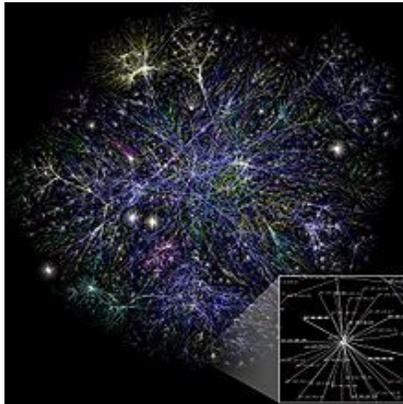


Gambar 5 : Ilustrasi Topologi Mesh

Setiap titik berkomunikasi dengan titik lainnya dengan cara saling mengirimkan paket. Paket tersebut berisi data dan informasi yang akan disampaikan kepada titik tujuan. Paket tersebut dikirim dari komputer pengirim, melewati media interkoneksi hingga sampai kepada komputer tujuan. Ada kalanya, paket ini melewati beberapa simpul dan beberapa lintasan dalam perjalanannya, sehingga sebuah simpul harus memiliki minimal sebuah lintasan dengan simpul tujuan jika ingin berkomunikasi.

Jaringan komputer sangat beragam, mulai dari skala kecil hingga skala besar. Jaringan skala kecil adalah sebuah jaringan LAN dan jaringan Wireless LAN, seperti jaringan access point LABDAS III IF, atau jaringan access point hotspot Selasar Jaringan skala menengah

adalah sebuah jaringan yang terdiri dari banyak LAN, dimana setiap LAN tersebut merujuk kepada sebuah titik yang lebih tinggi, yang menjadi pusat dari setiap LAN tersebut, contohnya adalah jaringan komputer di ITB. Jaringan skala besar adalah jaringan yang daerah kerjanya sangat besar, seperti antar kota, negara, bahkan antar benua. Contoh jaringan skala besar adalah Intercontinental Network (internet) yang memiliki milyaran titik dan milyaran interkoneksi, dan melibatkan banyak macam media interkoneksi.



Gambar 6 : Ilustrasi Graf dari jaringan internet.

Semakin besar skala jaringan tersebut, semakin rumit juga cara mengatur dan tingkat kompleksitasnya. Sehingga muncullah teknik-teknik tertentu untuk mengatur jaringan tersebut. Kebanyakan teknik-teknik tersebut mengatur interkoneksi antar titik dan mengatur komunikasi dan paket dari titik-titik. Salah satu yang akan dibahas dalam paper ini adalah teknik routing.

III. ROUTING

Setiap titik dalam setiap jaringan memiliki sebuah alamat, yang dikenal dengan alamat IP (IP address). Dimana setiap IP bersifat unik dan tidak boleh sama. Setiap titik yang terhubung dalam jaringan akan berkomunikasi dengan titik yang lain dengan cara mengirim paket dimana dalam setiap paket terdapat alamat IP pengirim dan alamat IP tujuan. Paket ini harus diarahkan agar dapat sampai kepada titik tujuan. Proses pengarahan paket inilah yang dinamakan Routing.

Routing adalah proses memilih lintasan yang akan ditempuh oleh sebuah paket dalam sebuah jaringan komputer untuk mengirim lalu lintas jaringan. Alat yang digunakan dalam proses Routing ini dinamakan Router. Routing ini dilakukan dalam banyak macam jaringan, seperti jaringan telepon, jaringan internet, dan jaringan transportasi.

Dalam proses routing ini, sebuah jaringan digambarkan sebagai sebuah graf berbobot dimana setiap interkoneksi antar titik dalam jaringan memiliki value tertentu. Value

ini terdiri dari bandwidth, network delay, hop count, path cost, load, reliability, dan biaya komunikasi. Setiap router harus mencari rute dengan biaya paling rendah.

Proses pencarian ini dapat diibaratkan seperti mencari lintasan dengan bobot terkecil dari sebuah graf berbobot. Tetapi, langkah-langkahnya tidak sesederhana itu mengingat router menangani sebuah graf dengan banyak titik. Sehingga router pun juga bermacam-macam, dari skala kecil hingga skala besar, bergantung kepada kompleksitas jaringan yang ditangani.



Gambar 7 : Router Skala Kecil



Gambar 8 : Router skala besar

Dalam makalah ini, saya membahas tentang packet switching, dimana dalam packet switching ini, routing mengatur tentang packet forwarding (tentang bagaimana sebuah paket diarahkan ke titik lain sesuai dengan jalur yang telah dipilih). Biasanya, titik-titik tempat lewatnya paket ini adalah network hardware yang bisa berupa router, bridge, gateway, firewall, atau switch. Proses routing ini biasanya dilakukan berdasarkan routing tables, yang menyimpan catatan rute ke berbagai network tujuan. Routing table sendiri harus disusun secara akurat agar dapat digunakan. Cara menyusun routing table inilah yang bermacam-macam dan akan dibahas di makalah ini.

Dalam prakteknya, Routing Table dapat disusun secara statis (static routing / non-adaptive routing), atau secara dinamis (dynamic routing). Jaringan komputer skala kecil dapat menggunakan static routing, ketika network administrator nya masih sanggup untuk mendefinisikan

semua route dalam network tersebut. Namun untuk jaringan komputer skala besar, yang memiliki topologi lebih kompleks dan dapat berubah-ubah dengan cepat, static routing nampak tidak feasible, maka digunakan teknik dynamic routing/adaptive routing.

IV. ALGORITMA ROUTING

A. Algoritma Breadth-First Search (BFS)

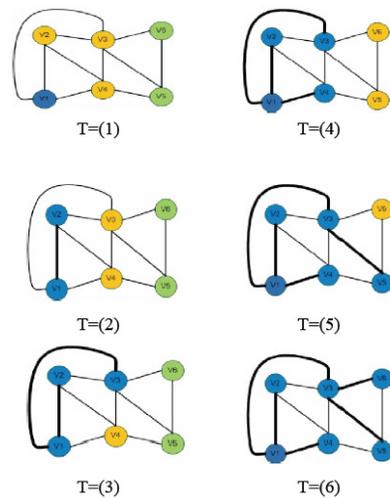
Routing dapat memanfaatkan algoritma BFS. Dimana algoritma ini biasanya digunakan ketika teknik routing yang digunakan adalah static routing.

Algoritma breadth-first adalah algoritma pencarian yang melakukan pencarian secara melebar, mengunjungi semua simpul dalam sebuah graf secara preorder. Algoritma ini mengunjungi sebuah simpul, lalu mengunjungi semua simpul yang bertetangga dengan simpul tersebut. Lalu mengunjungi simpul-simpul yang bertetangga dengan tetangga dari simpul tersebut. Dengan kata lain, jika sebuah graf tersebut adalah pohon berakar, maka algoritma akan mengunjungi semua simpul pada level 1, selanjutnya level 2, dan seterusnya hingga semua simpul selesai dikunjungi atau ketika apa yang dicari sudah ditemukan.

Algoritma ini menggunakan struktur data queue, dimana queue itu digunakan untuk menyimpan simpul-simpul yang telah dikunjungi, untuk digunakan sebagai acuan untuk mengunjungi simpul-simpul yang bertetangga dengan simpul-simpul yang telah dikunjungi tersebut.

Secara umum, langkah-langkah algoritma Breadth-first adalah sebagai berikut :

1. Menentukan simpul yang menjadi akar (misal dinamakan simpul x).
2. Melakukan kunjungan terhadap simpul-simpul yang bertetangga dengan simpul x (level 1).
3. Melakukan kunjungan terhadap simpul-simpul yang bertetangga dengan simpul level 1, tetapi bukan akar.
4. Melakukan kunjungan terhadap simpul-simpul yang bertetangga dengan simpul level 2, tetapi bukan merupakan simpul level 1
5. Untuk setiap simpul yang dikunjungi, diperiksa apakah merupakan simpul tujuan atau bukan. Jika merupakan simpul tujuan, maka searching berakhir, jika bukan maka searching akan terus berlanjut hingga semua simpul dikunjungi atau hingga simpul tujuan ditemukan.



Gambar 9 : Skema pencarian breadth-first

Dalam gambar, simpul berwarna biru merepresentasikan simpul yang sudah dikunjungi. Dan garis tebal merupakan arah penelusuran pada saat pengunjungan simpul.

Algoritma breadth-first ini dilakukan untuk mencari route yang memungkinkan untuk menghubungkan sebuah titik ke sebuah titik yang lain, atau dalam representasi graf, menghubungkan dari sebuah simpul ke simpul yang lain melalui sisi-sisi yang ada. Sisi-sisi ini baru dianggap valid ketika kedua node terhubung secara fisik dalam dunia nyata (bisa wired maupun wireless), dan network administrator mendefinisikan route antar kedua titik tersebut.

B. Algoritma Dynamic Programming

Dynamic programming adalah cara memecahkan masalah dengan cara menguraikan solusi menjadi sekumpulan tahapan (stage) sedemikian sehingga solusi dari persoalan dapat dipandang dari serangkaian keputusan yang saling berkaitan. Biasanya, solusi yang optimal dihitung menggunakan tabel-tabel.

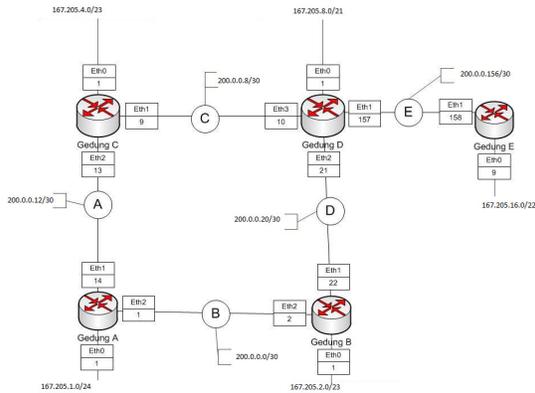
Karakteristik penyelesaian persoalan dengan program dinamis :

- Terdapat sejumlah berhingga pilihan yang mungkin
- Solusi pada setiap tahap dibangun dari hasil solusi tahap sebelumnya
- Menggunakan persyaratan optimasi dan kendala untuk membatasi sejumlah pilihan yang harus dipertimbangkan pada suatu tahap
- Prinsip optimalitas : rangkaian keputusan yang optimal dibuat dengan menggunakan prinsip dimana jika solusi total optimal, maka bagian solusi sampai tahap ke-k juga optimal.

Dalam network routing, Dynamic Programming ini digunakan dalam metode Dynamic Routing / Adaptive Routing. Seluruh rangkaian jaringan dipecah-pecah

menjadi bagian-bagian kecil yang dinamakan subnet. Setiap subnet terhubung keluar subnet melalui sebuah router. Seluruh router yang ada dalam jaringan, direpresentasikan sebagai sebuah simpul, dan interkoneksi antar router dipandang sebagai sebuah sisi. Penghitungan solusi optimal dihitung dan disimpan dalam routing table.

Sebagai contoh, ada sebuah kampus dengan 5 gedung, yang masing-masing gedung dihubungkan ke dalam jaringan kampus melalui sebuah router. gambarnya adalah sebagai berikut.



Gambar 10 : Contoh topologi sebuah jaringan komputer kampus

Setiap router dipandang sebagai sebuah simpul. dan setiap collision network (A,B,C,D,E) dipandang sebagai sebuah sisi. Pertama-tama, setiap router mengukur cost nya dengan setiap simpul yang terhubung dengan router tersebut. Misal, router gedung A menghitung cost ke router gedung B dan C, router gedung E menghitung cost ke gedung D, dst. Kemudian, setiap router memberikan informasi yang dimiliki ke tetangganya. Sehingga router gedung A memiliki informasi route ke router gedung D, router gedung E memiliki informasi route ke router gedung C, dst. Proses tukar informasi ini diulang secara terus menerus setiap selang waktu tertentu dan setiap kali ada router baru yang terhubung ke jaringan tersebut, sehingga informasi akan terus terbaharui dan seluruh komponen dalam jaringan akan selalu terhubung.

Ketika setiap cost dari semua simpul ke simpul tetangganya sudah diketahui, maka dynamic programming sudah dapat dilakukan. Jika router gedung A ingin menghubungi router gedung E, dia tinggal membandingkan (cost dari router gedung B ke router gedung E + cost dari router gedung A ke router gedung B) dengan (cost dari router gedung C ke router gedung E + cost dari router gedung A ke router gedung C), dan berlaku untuk interkoneksi antar router yang lain.

V. CONTOH IMPLEMENTASI

Dalam makalah ini, saya akan memberikan contoh implementasi dari algoritma dynamic routing yang telah saya bahas sebelumnya. Implementasi ini dilakukan dengan menggunakan software Netkit yang dilakukan di

dalam lingkungan Sistem Operasi Linux Ubuntu 10.04. Dalam implementasi ini, saya mencoba melakukan simulasi pengelolaan sebuah network yang digambarkan pada gambar 10.

Pertama-tama, setiap router dinyalakan, secara otomatis setiap router akan mengukur cost antara router tersebut dengan router tetangganya. Kemudian, setiap router akan bertukar informasi dan akan terus memperbaharui routing table yang dimiliki.

Berikut adalah hasil screenshot dari routing table akhir yang dimiliki oleh router gedung D :

```
rD:~# route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
200.0.0.0 200.0.0.22 255.255.255.252 UG 2 0 0 eth2
200.0.0.12 200.0.0.9 255.255.255.252 UG 2 0 0 eth3
200.0.0.8 * 255.255.255.252 U 0 0 0 eth3
200.0.0.20 * 255.255.255.252 U 0 0 0 eth2
200.0.0.156 * 255.255.255.252 U 0 0 0 eth1
167.205.1.0 200.0.0.9 255.255.255.0 UG 3 0 0 eth3
167.205.4.0 200.0.0.9 255.255.254.0 UG 2 0 0 eth3
167.205.2.0 200.0.0.22 255.255.254.0 UG 2 0 0 eth2
167.205.16.0 200.0.0.158 255.255.252.0 UG 2 0 0 eth1
167.205.8.0 * 255.255.248.0 U 0 0 0 eth0
rD:~#
```

Gambar 11 : Hasil Routing Table dari Router D

Dari router tersebut, dapat dilihat route-route yang dipilih untuk mencapai titik-titik tertentu. Titik di gedung A misalnya (167.205.1.0, Genmask 255.255.255.0), dapat diraih dengan melewati router dengan alamat IP 200.0.0.9 (router gedung C), beserta cost yang diperlukan (metric=3). Misalkan lagi titik di gedung B (167.205.2.0 Genmask 255.255.254.0), dapat diraih dengan melewati router dengan alamat Ip 200.0.0.22 (Router gedung B), beserta cost nya (metric=2).

Berikut adalah hasil screenshot dari router-router yang lain.

```
rA:~# route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
200.0.0.0 * 255.255.255.252 U 0 0 0 eth2
200.0.0.12 * 255.255.255.252 U 0 0 0 eth1
200.0.0.8 200.0.0.13 255.255.255.252 UG 2 0 0 eth1
200.0.0.20 200.0.0.2 255.255.255.252 UG 2 0 0 eth2
200.0.0.156 200.0.0.13 255.255.255.252 UG 3 0 0 eth1
167.205.1.0 * 255.255.255.0 U 0 0 0 eth0
167.205.4.0 200.0.0.13 255.255.254.0 UG 2 0 0 eth1
167.205.2.0 200.0.0.2 255.255.254.0 UG 2 0 0 eth2
167.205.16.0 200.0.0.13 255.255.252.0 UG 4 0 0 eth1
167.205.8.0 200.0.0.13 255.255.248.0 UG 3 0 0 eth1
rA:~#
```

Gambar 12 : Hasil Routing Table dari Router A

```
rB:~# route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
200.0.0.0 * 255.255.255.252 U 0 0 0 eth2
200.0.0.12 200.0.0.1 255.255.255.252 UG 2 0 0 eth2
200.0.0.8 200.0.0.21 255.255.255.252 UG 2 0 0 eth1
200.0.0.20 * 255.255.255.252 U 0 0 0 eth1
200.0.0.156 200.0.0.21 255.255.255.252 UG 2 0 0 eth1
167.205.1.0 200.0.0.1 255.255.255.0 UG 2 0 0 eth2
167.205.4.0 200.0.0.1 255.255.254.0 UG 3 0 0 eth2
167.205.2.0 * 255.255.254.0 U 0 0 0 eth0
167.205.16.0 200.0.0.21 255.255.252.0 UG 3 0 0 eth1
167.205.8.0 200.0.0.21 255.255.248.0 UG 2 0 0 eth1
rB:~#
```

Gambar 13 : Hasil Routing Table dari Router B

```

rC:~# route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
200.0.0.0 200.0.0.14 255.255.255.252 UG 2 0 0 eth2
200.0.0.12 * 255.255.255.252 U 0 0 0 eth2
200.0.0.8 * 255.255.255.252 U 0 0 0 eth1
200.0.0.20 200.0.0.10 255.255.255.252 UG 2 0 0 eth1
200.0.0.156 200.0.0.10 255.255.255.252 UG 2 0 0 eth1
167.205.1.0 200.0.0.14 255.255.255.0 U 2 0 0 eth2
167.205.4.0 * 255.255.254.0 U 0 0 0 eth0
167.205.2.0 200.0.0.14 255.255.254.0 UG 3 0 0 eth2
167.205.16.0 200.0.0.10 255.255.252.0 UG 3 0 0 eth1
167.205.8.0 200.0.0.10 255.255.248.0 UG 2 0 0 eth1
rC:~#

```

Gambar 14 : Hasil Routing Table dari Router C

```

rE:~# route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
200.0.0.0 200.0.0.157 255.255.255.252 UG 3 0 0 eth1
200.0.0.12 200.0.0.157 255.255.255.252 UG 3 0 0 eth1
200.0.0.8 200.0.0.157 255.255.255.252 UG 2 0 0 eth1
200.0.0.20 200.0.0.157 255.255.255.252 UG 2 0 0 eth1
200.0.0.156 * 255.255.255.252 U 0 0 0 eth1
167.205.1.0 200.0.0.157 255.255.255.0 UG 4 0 0 eth1
167.205.4.0 200.0.0.157 255.255.254.0 UG 3 0 0 eth1
167.205.2.0 200.0.0.157 255.255.254.0 UG 3 0 0 eth1
167.205.16.0 * 255.255.252.0 U 0 0 0 eth0
167.205.8.0 200.0.0.157 255.255.248.0 UG 2 0 0 eth1
rE:~#

```

Gambar 15 : Hasil Routing Table dari Router E

Selanjutnya, saya mencoba mengetahui jalan mana yang ditempuh, ketika sebuah komputer di gedung A, akan menghubungi komputer yang ada di gedung E, dengan menggunakan perintah traceroute. Dan hasilnya adalah sebagai berikut :

```

pcA1:~# ping 167.205.16.10
PING 167.205.16.10 (167.205.16.10) 56(84) bytes of data:
64 bytes from 167.205.16.10: icmp_seq=1 ttl=60 time=14.1 ms
64 bytes from 167.205.16.10: icmp_seq=2 ttl=60 time=3.09 ms
64 bytes from 167.205.16.10: icmp_seq=3 ttl=60 time=3.59 ms
64 bytes from 167.205.16.10: icmp_seq=4 ttl=60 time=3.22 ms
64 bytes from 167.205.16.10: icmp_seq=5 ttl=60 time=2.83 ms
^C
--- 167.205.16.10 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4038ms
rtt min/avg/max/mdev = 2.835/5.388/14.188/4.407 ms
pcA1:~# traceroute 167.205.16.10
traceroute to 167.205.16.10 (167.205.16.10), 30 hops max, 60 byte packets
 1 (167.205.1.1) 0.617 ms 0.664 ms 0.703 ms
 2 (200.0.0.13) 1.344 ms 1.063 ms 1.413 ms
 3 (200.0.0.10) 2.462 ms 2.125 ms 1.875 ms
 4 (200.0.0.158) 2.370 ms 2.306 ms 3.130 ms
 5 (167.205.16.10) 3.088 ms 3.020 ms 2.902 ms
pcA1:~#

```

Gambar 16 : Hasil ping dari komputer di gedung A ke komputer di gedung E

Dari traceroute di atas, dapat diketahui bahwa komunikasi dari komputer di gedung A ke komputer di gedung E dilakukan melewati titik-titik :

1. 167.205.1.1 (Router gedung A)
2. 200.0.0.13 (Router gedung C)
3. 200.0.0.10 (Router gedung D)
4. 200.0.0.158 (Router gedung E)
5. 167.205.16.10 (Komputer tujuan di gedung E)

Langkah-langkah tersebut diambil berdasarkan routing table di setiap router yang ada di setiap gedung.

VI. KESIMPULAN

Pengelolaan jaringan komputer dapat menggunakan

strategi algoritma untuk memecahkan masalah routing. Untuk metode static routing, dapat menggunakan algoritma BFS biasa, dimana seluruh rute dari setiap titik ke titik tetangganya sudah didefinisikan sebelumnya. Untuk metode dynamic routing dapat menggunakan algoritma dynamic programming.

Untuk jaringan komputer skala kecil, metode static routing dengan algoritma BFS lebih optimal karena lebih terjamin interkoneksinya sehingga network admin dapat membagi beban dari masing-masing router secara spesifik.

Untuk jaringan komputer skala besar, metode dynamic routing dengan algoritma dynamic programming lebih optimal karena dapat menangani perubahan titik-titik pada jaringan tersebut secara realtime.

DAFTAR REFERENSI

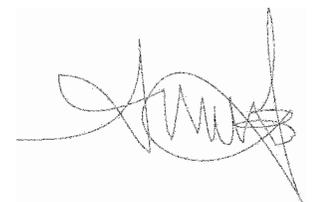
- [1] http://en.wikipedia.org/wiki/Graph_theory. Waktu Akses : 10 Desember 2012.
- [2] http://en.wikipedia.org/wiki/Computer_network . Waktu Akses : 11 Desember 2012.
- [3] <http://en.wikipedia.org/wiki/Routing> . Waktu Akses : 11 Desember 2012.
- [4] http://en.wikipedia.org/wiki/Breadth-first_search . Waktu Akses : 11 Desember 2012.
- [5] http://wiki.netkit.org/index.php/Labs_Official . Waktu Akses : 11 Desember 2012.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 17 Desember 2012

ttd



Arief Suharsono
13510087