

# IMPLEMENTASI PROTOKOL ROUTING *DESTINATION SEQUENCED DISTANCE-VECTOR (DSDV)* PADA JARINGAN WIRELESS NS-2 DENGAN NAM

Anjas Purnomo , Herman Kurniawan

Jurusan Teknik Informatika, Fakultas Teknologi Industri, Universitas Veteran Jawa Timur

Jl. Raya Rungkut Madya Gunung Anyar Surabaya

E-mail : [anjaspurnomo@yahoo.com](mailto:anjaspurnomo@yahoo.com), [hermanhatakekakashiro@yahoo.com](mailto:hermanhatakekakashiro@yahoo.com)

## ABSTRAKSI

Untuk membangun jaringan nirkabel untuk mengirim informasi dapat perkiraan dari data rate, daerah cakupan, topologi jaringan, ukuran jaringan, protokol *routing* dan konsumsi daya. Protokol jaringan nirkabel banyak macamnya, salah satunya adalah DSDV. DSDV merupakan salah satu *Proactive Routing Protocol* yang membutuhkan setiap *node* untuk mengirimkan paket *routing* update ke seluruh *node* tetangganya secara periodik. Setiap *node* menyimpan tabel *routing* yang mengandung informasi yang dibutuhkan untuk sampai ke *node* tujuan.

Penelitian ini melakukan simulasi menggunakan aplikasi Network Simulator 2 (NS-2) untuk menganalisis kinerja DSDV pada jaringan wireless dengan NAM sebagai ilustrasinya. Simulasi dibagi menjadi dua skenario untuk mengetahui pengaruh mobilitas *node* dan trafik terhadap kinerja DSDV. Skenarionya adalah selang kecepatan dan selang waktu dari NAM. Hasil simulasi menunjukkan kinerja DSDV dipengaruhi oleh skenario dimana semakin lama waktu dan semakin sedikit *step*nya, kinerja DSDV semakin tinggi. DSDV cocok untuk jaringan wireless berskala kecil dengan jumlah *node* yang sedikit.

**Kata kunci:** DSDV, jaringan wireless, Network Simulator 2, Network Animator.

## 1. PENDAHULUAN

Jaringan komputer telah mengalami perubahan teknologi dari menggunakan kabel untuk menghubungkan komputer-komputer menjadi wireless atau tanpa kabel. Jaringan wireless ad-hoc adalah jaringan tanpa kabel yang menghubungkan pasangan-pasangan *node* dan sangat tergantung dari jarak transmisi dari tiap-tiap *node* (Mukherjee et al.

2003). Protokol Destination-Sequenced Distance-Vector (DSDV) ialah salah satu protokol *routing* khusus jaringan ad hoc.

Protokol DSDV diukur kapabilitasnya dalam menghadapi perubahan topologi karena jumlah *node* dan besarnya trafik yang terjadi antar *node*.

## 2. DESTINATION SEQUENCED DISTANCE-VECTOR (DSDV)

DSDV merupakan algoritma *routing protocol* ad hoc proaktif yang didasari pada Bellman – Ford yang pertama kali dikenalkan, kontribusi algoritma ini adalah untuk mengatasi *Routing Loop*. Pada DSDV, digunakan *sequence number* untuk mengirimkan pesan pada jaringan. *Sequence number* dihasilkan juga saat ada perubahan dalam jaringan, hal ini terjadi karena sifat *table routing node* pada jaringan yang menggunakan protokol proaktif yang update secara periodik, serta *Triggered update* ulang digunakan oleh *node* untuk mengupdate *node* yang masuk dan keluar dari jaringan.

Dalam metode *routing* DSDV, setiap *node* yang berada dalam jaringan, akan memelihara tabel *routing* ke *node* tetangganya, tabel *routing* yang dimiliki oleh setiap *node* berisi tentang : alamat tujuan *node*, jumlah hop yang diperlukan untuk mencapai tujuan, serta sequenced number. Jika tabel *routing* dalam satu *node* telah diupdate, maka akan dipilih rute untuk mencapai *node* tujuan dengan beberapa pertimbangan sebagai berikut :

- Memiliki *sequence number* yang terbaru, hal ini dapat dilihat dari nilai sequenced number yang paling besar
- Jika nilai *sequence number* sama, maka akan dilihat nilai *metric*nya, nilai *metric* yang lebih

kecil akan dipilih.

Penerapan protocol DSDV memiliki kelebihan , yaitu :

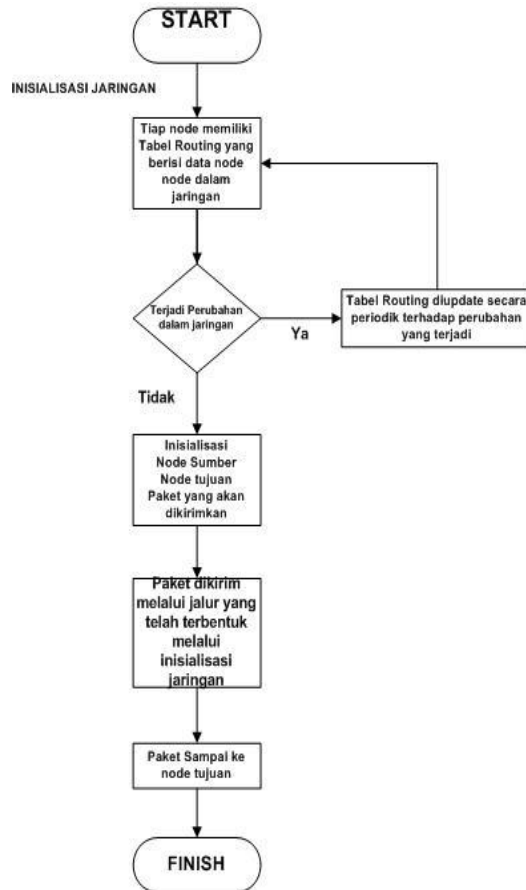
- DSDV menjamin tidak ada looping route
- DSDV dapat mereduksi masalah count to infinity
- DSDV dapat menghindari trafik lebih dengan kenaikan drastis update penuh untuk dump
- DSDV hanya maintenance path terbaik menuju tujuan, dari sekian banyak path ketujuan.

Keterbatasan yang dimiliki algoritma routing protocol DSDV ini menjadi kelemahannya, yaitu :

- Banyak memakan alokasi bandwidth untuk broadcasting informasi yang tidak terlalu berguna, meskipun tidak ada perubahan yang berarti pada topologi jaringan, hal ini dikarenakan sifatnya yang selalu update setiap waktunya.
- DSDV tidak mensupport multi path routing.
- Sangat sulit untuk mendeterminasi waktu delay saat broadcasting informasi ke semua node.
- Table routing pada node di jaringan DSDV sangat sulit untuk dimaintenance pada jaringan berskala besar, mengingat jumlah node yang banyak dan luas jaringan yang semakin diperluas (scalable) yang meningkatkan overhead.

Berikut merupakan proses secara keseluruhan algoritma routing protocol DSDV :

- Diawal transmisi sebelum dilakukan , tiap node memiliki table yang berisi data node node dalam jaringan.
- Jika terjadi perubahan topologi jaringan setelah paket data dikirimkan, table routing akan diupdate secara periodic.
- Jika tidak ada permasalahan pada topologi jaringan, maka node akan dikirimkan setelah node sumber dan node tujuan diinisialisasi, lalu paket akan sampai ke node tujuan.



Gambar 1: Diagram Alir proses kerja DSDV

### 3. LANGKAH-LANGKAH SIMULASI

Simulasi dijalankan pada komputer berprosesor Intel Core 2 Duo 2.20 GHz, memori 2Gb, sistem operasi windows XP di dalam VMware dan waktu yang dibutuhkan untuk menjalankan setiap skenario simulasi sekitar 8 menit. Langkah simulasi yang dilakukan sebagai berikut :

- Menjalankan VMware dengan sistem operasi Windows XP, dan meng-install serta menjalankan program Emulator Cygwin
- Menjalankan simulasi di NS-2 dengan skrip yang sudah dibuat dalam bahasa Otcl (*variant object oriented* dari Tcl). Yaitu di dalam file "anjas-herman.tcl"
- Informasi dari hasil tersebut kemudian di simulasikan dengan menjalankan Network Animator(NAM) dalam NS-2 kemudian melakukan skenario yang ada pada NAM

## 4. HASIL DAN PEMBAHASAN

### Hasil tanpa NAM

Setelah *source code* dijalankan di NS-2 dengan mengetikkan: `ns anjas-herman.tcl` maka hasilnya akan keluar text seperti gambar di bawah ini:

```
admin@upn-de8b0f1e26a /
$ cd ..

admin@upn-de8b0f1e26a /
$ cd c:

admin@upn-de8b0f1e26a /cygdrive/c
$ cd cygwin/home/admin/ns-allinone-2.29/ns-2.29/tcl/ex

admin@upn-de8b0f1e26a /cygdrive/c/cygwin/home/admin/ns-allinone-2.29/ns-2.29/tcl/ex
$ ns anjas-herman.tcl
num_nodes is set 3
warning: Please use -channel as shown in tcl/ex/wireless-mitf.tcl
INITIALIZE THE LIST xListHead
Mulai .....
channel.ccsendUp - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5, distCST_ = 550.0
SORTING LISTS ...DONE!
Selesai
```

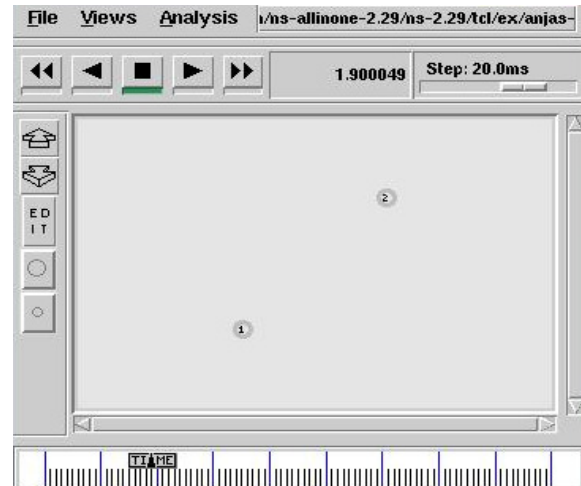
Gambar 2: Hasil running dari sourcode

Terlihat warning dikarenakan channel pada `wireless-mitf.tcl` karena *sourcode* ini sama dengan `wireless-mitf.tcl` hanya saja di beri protokol DSDV. Pada hasil tersebut juga terlihat `highestAntennaZ_ = 1,5` dan `distCST_ = 550,0`

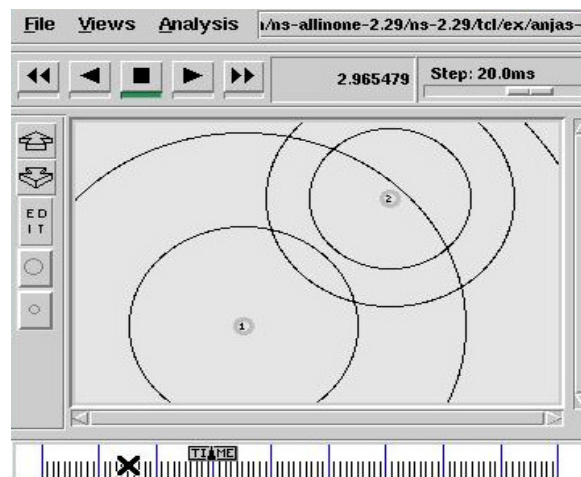
### Hasil dengan NAM

Hasil *source code* pada NAM dengan kecepatan `step: 20.0m/s` dan waktu sekitar 15 detik terlihat menunjukkan hasil yang belum terlihat sinyal antar node dengan hasil 1,900049. Setelah waktu sekitar 25 detik sinyal sudah terlihat dengan hasil 2.965479.

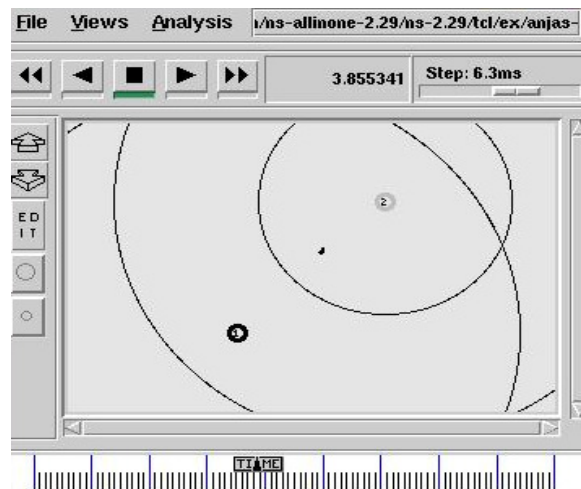
Setelah waktu berjalan menjadi 30 detik dan `step` diubah menjadi 6.3m/s maka hasil berubah lebih tinggi menjadi 3.855341. Setelah beberapa waktu maka di akhir *timeline* dengan `step` yang sama hasil menjadi naik drastis yaitu 8.997158. Hal ini memperlihatkan semakin lama waktu dan semakin sedikit *step*nya, kinerja DSDV semakin tinggi. DSDV cocok untuk jaringan wireless berskala kecil dengan jumlah node yang sedikit.



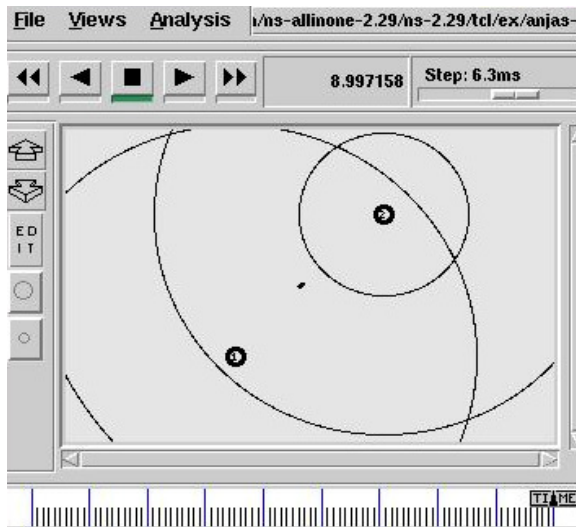
Gambar 3: Hasil NAM dengan step: 20.0m/s, waktu= 15



Gambar 4: Hasil NAM dengan step: 20.0m/s, waktu= 25



Gambar 5: Hasil NAM dengan step: 20.0m/s, waktu= 25



Gambar 6: Hasil NAM dengan step: 20.0m/s, waktu= 25

## 5. KESIMPULAN DAN SARAN

### Kesimpulan

Jaringan wireless ad hoc ialah jaringan nirkabel tanpa infrastruktur dimana node-node di dalamnya bertindak sebagai router. Agar node-node dapat saling berkomunikasi dengan baik satu sama lain, dibutuhkan protokol routing yang dapat mengantisipasi pergerakan dan trafik antar node. Pada penelitian ini, dilakukan simulasi menggunakan aplikasi NS-2 untuk menganalisis kinerja DSDV pada jaringan wireless dengan NAM.

DSDV sangat dipengaruhi oleh waktu dan kecepatan/step juga mobilitas node, seperti terlihat hasil pada NAM. Kinerja DSDV meningkat seiring dengan semakin lama waktu dan semakin sedikit step. DSDV juga kurang mampu menangani jaringan berskala besar, terlihat oleh karena itu pada source code hanya menyajikan 2 node.

### Saran

Protokol pada penelitian ini hanya DSDV, diharapkan untuk penelitian selanjutnya diteliti juga protokol-protokol routing yang lain, seperti DSR atau TORA. Diharapkan juga untuk penelitian selanjutnya skala jaringan yang lebih luas dan jenis trafik yang lebih beragam.

## 6. DAFTAR PUSTAKA

- [1] <http://anakkebo.wordpress.com/2011/09/11/routing-protokol-destination-sequenced-distance-vector-dsdv/>
- [2] <http://digilib.its.ac.id/public/ITS-Undergraduate-15901-Paper-582323.pdf>
- [3] <http://digilib.its.ac.id/public/ITS-Undergraduate-6878-2203100057-ANALISIS%20PERBANDINGAN%20ROUTING%20PROTOKOL%20DSDV%20DAN%20AODV%20PADA%20JARINGAN%20AD%20HOC.pdf>
- [4] [http://eprints.undip.ac.id/27362/1/L2F005585\\_MTA.pdf](http://eprints.undip.ac.id/27362/1/L2F005585_MTA.pdf)
- [5] [http://journal.ui.ac.id/upload/artikel/02\\_RiriFS\\_ANALISIS%20KINERJA%20PROTOKOL.pdf](http://journal.ui.ac.id/upload/artikel/02_RiriFS_ANALISIS%20KINERJA%20PROTOKOL.pdf)
- [6] <http://repository.ipb.ac.id/bitstream/handle/123456789/14523/G2007mfr.pdf?sequence=6>
- [7] <http://www.eepis-its.edu/uploadta/downloaddmk.php?id=1203>

## 7. BIODATA PENULIS



**Anjas Purnomo** lahir di Bogor. Sekarang sedang menyelesaikan pendidikan strata satu di Universitas Pembangunan Nasional "Veteran" Jawa Timur.



**Herman Kurniawan** lahir di Surabaya. Sekarang sedang menyelesaikan pendidikan strata satu di Universitas Pembangunan Nasional "Veteran" Jawa Timur.

## SOURCE CODE :

### #nama file : anjas-herman.tcl

```
set val(jumlahnode) 3
set val(panjang)    200
set val(lebar)       200

set ns_              [new Simulator]

set jejak             [open anjas-herman.tr w]

$nns_ trace-all $jejak

set namtrace [open anjas-herman.nam w]

$nns_ namtrace-all-wireless $namtrace $val(panjang) $val(lebar)
```

### #pembuatan topografi

```
set daerah           [new Topography]

$daerah load_flatgrid $val(panjang) $val(lebar)

set god_              [create-god $val(jumlahnode)]
```

### #konfigurasi node

```
$nns_ node-config      -adhocRouting DSDV -llType LL \
                        -macType Mac/802_11 \
                        -ifqType Queue/DropTail/PriQueue \
                        -ifqLen 50 -antType Antenna/OmniAntenna \
                        -propType Propagation/TwoRayGround \
                        -phyType Phy/WirelessPhy \
                        -channelType Channel/WirelessChannel \
                        -topoInstance $daerah -agentTrace OFF \
                        -routerTrace OFF -macTrace ON -movementTrace ON \
                        -txPower 0.0075 -rxPower 1
```

### #membuat node baru

```
set ap [$nns_ node]
set n0 [$nns_ node]
set n1 [$nns_ node]
```

### #mengambil nilai MAC

```
set mac_ap [$ap getMac 0]
set mac_n0 [$ap getMac 0]
set mac_n1 [$ap getMac 0]
```

### #node ap sebagai AP

```
set Alamat_AP [$mac_ap id]
```

### #semua node diberi tahu base station ID

```
$mac_ap bss_id $Alamat_AP
$mac_n0 bss_id $Alamat_AP
$mac_n1 bss_id $Alamat_AP
```

### #gerakan acak ditiadakan

```
$ap random-motion 0
$n0 random-motion 0
```

```
$n1 random-motion 0
```

### #penentuan posisi masing2 node

```
$ap set X_ 100.0
$ap set Y_ 100.0
$ap set Z_ 0.0
$n0 set X_ 50.0
$n0 set Y_ 50.0
$n0 set Z_ 0.0
$n1 set X_ 150.0
$n1 set Y_ 150.0
$n1 set Z_ 0.0
```

### # perpindahan node

```
$nns_ at 3.0 "$n0 setdest 55.0 40.0 0.0"
$nns_ at 6.0 "$n0 setdest 65.0 30.0 0.0"
$nns_ at 4.0 "$n1 setdest 125.0 120.0 0.0"
$nns_ at 7.0 "$n1 setdest 130.0 160.0 0.0"
```

### #membuat agent UDP dengan trafik CBR n0 sebagai pengirim n1 sebagai penerima

```
set udp0 [new Agent/UDP]
$udp0 set class_ 0
$nns_ attach-agent $n0 $udp0
set cbr0 [new Application/Traffic/CBR]
$cbr0 set packetSize_ 500
$cbr0 set interval_ 0.005
$cbr0 attach-agent $udp0
set null1 [new Agent/Null]
$nns_ attach-agent $n1 $null1
$nns_ connect $udp0 $null1
```

### #penjadwalan aliran trafik CBR

```
$nns_ at 2.0 "$cbr0 start"
$nns_ at 9.0 "$cbr0 stop"
```

### #reset semua node

```
$nns_ at 10.0 "$ap reset"
$nns_ at 10.0 "$n0 reset"
$nns_ at 10.0 "$n1 reset"
```

### #panggil produser stop

```
$nns_ at 11.0 "stop"
$nns_ at 10.0 "puts \"Selesai\""
```

```
proc stop { } {
    global ns_ jejak
    $nns_ flush-trace
    close $jejak
    exit 0 }
```

```
puts "Mulai ....."
```

```
$nns_ run
```