

# **RANCANGAN PROGRAM SIMULASI PENENTUAN LETAK LOKASI ANTENA TERBAIK MENGUNAKAN ALGORITMA FLETCHER-POWELL**

**Ngarap Im. Manik**

Mathematics & Statistics Department, School of Computer Science, Binus University  
Jl. K.H. Syahdan No. 9, Palmerah, Jakarta Barat 11480  
manik@binus.edu

## **ABSTRACT**

*To meet the needs of information delivery to be rapid, mobile and safe needed a model calculation of the best antenna location and utilization of transmission media. One model that can be used takes advantages of wireless as well as models Fletcher-Powell algorithm, which combines mathematics with information technology. The algorithm can calculate the best position of a transmitter with reference of maximum received intensity amount in all the Main Point Coverage. To simplify the calculation process a computer program is developed written in Borland C. the program evaluation results show that by utilizing this application a company can place transmitters better and more effectively.*

**Keywords:** *Main point coverage, Fletcher-Powell, wireless*

## **ABSTRAK**

*Untuk memenuhi kebutuhan penyampaian informasi yang cepat, mobile dan aman diperlukan suatu model perhitungan letak lokasi antena terbaik dan pemanfaatan media transmisinya. Salah satu model yang dapat digunakan memanfaatkan nirkabel serta model algoritma Fletcher-Powell yang mengkombinasikan matematika dengan teknologi informasi. Algoritma ini dapat menghitung posisi terbaik sebuah pemancar dengan acuan jumlah intensitas yang diterima di semua Main Coverage Point mencapai maksimum. Untuk memudahkan proses perhitungannya telah dikembangkan suatu program komputer yang ditulis dalam Borland C. Hasil uji coba program ini menunjukkan bahwa dengan aplikasi ini suatu perusahaan dapat meletakkan pemancar-pemancarnya dengan lebih baik dan efektif.*

**Kata kunci:** *Main coverage point, fletcher-powell, nirkabel*

## PENDAHULUAN

Seiring dengan perkembangan teknologi informasi dan komunikasi (TIK) saat ini, penyampaian informasi harus memenuhi kebutuhan mobilitas transmisi yang cukup kuat. Hal ini dapat dilakukan dengan menggunakan nirkabel. Karena tidak menggunakan kabel, teknologi ini menjadi sangat fleksibel sehingga dapat menyampaikan suatu informasi ke objek yang sedang bergerak. Contohnya adalah radio, antena televisi, telepon selular, dan lain-lain.

Teknologi informasi nirkabel sudah mulai dikembangkan secara besar-besaran, terutama informasi yang disampaikan melalui pemancar seperti radio dan telepon selular. Oleh karena itu, satu hal yang perlu pada sistem ini adalah bagaimana posisi letak optimal pemancar-pemancar di berbagai tempat agar informasi dapat dikirim dengan baik ke objek yang dituju.

Pada sisi lain muncul berbagai masalah sebagai dampak lanjutan dari penggunaan pemancar. Salah satu contoh dari masalah tersebut adalah pemancar yang akan didirikan bersifat permanen dan tidak bisa dipindah-pindahkan. Jadi untuk meletakkan suatu pemancar perlu dilakukan perhitungan dan pertimbangan yang cukup mendalam karena posisi pemancar inilah yang nantinya akan menentukan kesempurnaan dalam menyampaikan informasi-informasi yang akan dikirimkan.

Artikel ini membahas tentang optimasi sebuah fungsi jarak yang didapat dengan menjumlahkan semua jarak koordinat posisi target dengan koordinat posisi pemancar, agar didapat letak yang paling optimal. Pengoptimalan ini dilakukan dengan menggunakan metode *Fletcher-Powell* yang didalamnya terdapat iterasi-iterasi yang menggunakan batas toleransi. Dengan metode ini akan didapatkan hasil berupa koordinat optimal peletakan pemancar yang terbaik, sehingga diharapkan dari posisi tersebut dapat mengoptimalkan fungsi kerja dari pemancar tersebut. Untuk mempermudah perhitungan koordinat posisi yang dimaksud dilakukan dengan bantuan program komputer Borland C.

### Landasan Teori

Antena adalah bagian vital dari suatu pemancar atau penerima yang berfungsi untuk memancarkan atau menerima gelombang radio (yang berupa gelombang elektromagnetik) tanpa menggunakan struktur penuntun seperti kabel (kabel tembaga maupun serat optik). Antena mengubah getaran listrik dari radio menjadi getaran elektromagnetik yang disalurkan melalui udara. Ukuran fisik dari radiasinya akan setara dengan panjang gelombangnya. Semakin tinggi frekuensinya, antenanya akan semakin kecil. Kedua perangkat radio harus bekerja di frekuensi yang sama, dan *antena* akan melakukan pekerjaan bersamaan, mengirim dan menerima sinyal.

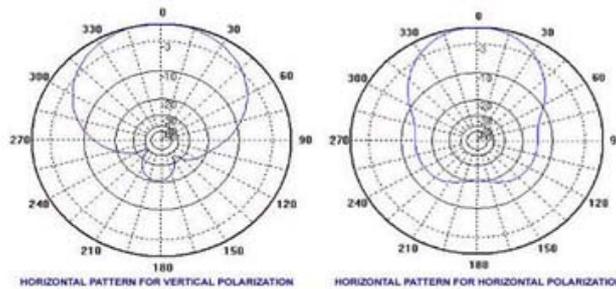
Antena yang ideal akan meradiasikan gelombang radio kesegala arah. Antena yang ideal disebut sebagai antena isotropis. Sebagai gambaran, jika antena isotropis diletakkan pada titik pusat dari bola maka antena isotropis akan mengisi semua ruang yang ada pada bola tersebut dengan radiasi gelombang radio (Kanginan, 2000).

### Pola radiasi dan bentuk-bentuk *antenna*

Bentuk antena bermacam-macam sesuai dengan desain, pola penyebaran dan frekuensi. Jenis antena yang akan dipasang harus sesuai dengan sistem yang akan dibangun, juga disesuaikan dengan kebutuhan penyebaran radiasinya. Dalam sistem *wireless*, antena digunakan untuk mengkonversi gelombang listrik menjadi gelombang elektromagnetik.

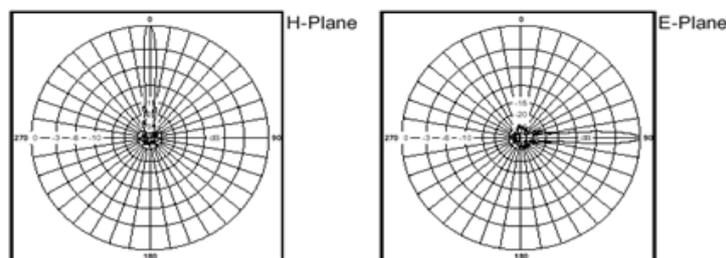
Berdasarkan pola radiasinya antena dibagi atas dua jenis, yaitu *directional* dan *omni directional*. Antena *directional* mempunyai radiasi terkuat pada arah tertentu, sementara radiasi pada

arah lainnya sangat kecil atau bahkan nol. Antena jenis ini merupakan jenis antena dengan *narrow beamwidth*, yaitu mempunyai sudut pemancaran yang kecil dengan daya lebih terarah, jaraknya jauh dan tidak bisa menjangkau area yang luas. Antena *directional* dibutuhkan jika arah pancaran atau penerimaan menuju ke arah tertentu. Antena televisi termasuk jenis ini karena letak rumah dan pemancar siaran TV tetap. Selain itu, diharapkan sinyal yang diterima pada arah tersebut jauh lebih besar daripada sinyal yang datang dari arah lain. Contoh antena *directional* adalah antena Yagi dan antena parabola. Antena Yagi sangat cocok untuk jarak pendek dan mempunyai gain antara 7 s/d 15 dBm. Berikut pola radiasi dari antena Yagi (Gambar 1):



Gambar 1. Pola radiasi antena Yagi (Sumber: situs Gstatic.com).

Antena parabola sangat cocok untuk jarak menengah atau jarak jauh. Mempunyai *gain* antara 18 sampai 28 dBm. Pola radiasi dari antena parabola adalah sebagai berikut (Gambar 2):



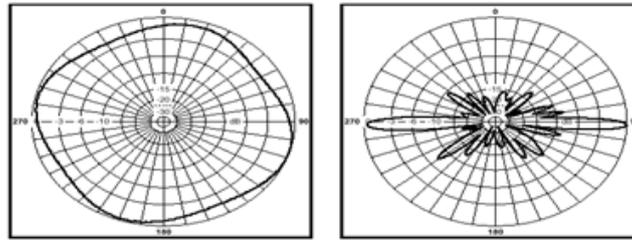
Gambar 2. Pola radiasi antena parabola (Sumber: situs Teknikcall-yogya.com).

Antena *omnidirectional* mempunyai pola radiasi yang sama ke segala arah horizontal. Antena ini mempunyai sudut pancaran yang besar (*wide beamwidth*) yaitu  $360^0$  dengan daya lebih meluas, jarak yang lebih pendek tetapi dapat melayani area yang luas. *Antenna omnidirectional* dibutuhkan jika pancaran atau penerimaan gelombang radio yang diharapkan datang dari segala arah. *Antenna-antenna* semacam ini cocok dipasang pada stasiun *broadcast* yang terletak di tengah kota dan berharap pancarannya dapat diterima di segala arah. Antena ini juga cocok dipasang pada mobil yang arah pergerakannya cepat berubah. Telepon seluler juga membutuhkan antena ini. Antena Omni biasanya dipakai oleh radio base untuk daerah pelayanan yang luas. Gainnya berkisar antara 3 sampai 10 dBm (Kanginan, 2000).

## Impedansi antena

Impedansi pada suatu titik di elemen antena adalah perbandingan antara tegangan terhadap arus di titik itu. Nilainya tergantung dari sifat resistif, kapasitif, induktif, dan frekuensi yang

digunakan. Satuan yang digunakan adalah Ohm, yang disimbolkan dengan Z. Transfer energi dari pemancar ke antenna lewat kabel dapat berlangsung secara efisien (tidak ada energi yang terbuang atau terpantul), impedansi antenna, kabel, dan pemancar harus sesuai. Efek terburuk dari impedansi yang tidak sama adalah timbulnya daya pantul (*reflected power*) dari antenna. Daya pantul yang kembali ke pemancar akan merusak rangkaian pemancar.



Gambar 3. Pola radiasi antenna omni (Sumber: situs Teknikcall-yogya.com).

Agar tidak terjadi kerusakan, perlu dilakukan penyesuaian impedansi (*impedansi matching*) pada antenna dan kabel sehingga sesuai dengan impedansi pemancar. Nilai *Voltage Standing Wave Ratio* (VSWR) mengindikasikan seberapa baik penyesuaian impedansi yang dilakukan. VSWR atau SWR yang tinggi menunjukkan bahwa sinyal yang dipantulkan masih lebih besar daripada sinyal yang dipancarkan antenna.

### Gain (penguatan antenna)

*Gain antenna* adalah perbandingan antara daya yang dipancarkan oleh suatu antenna dan daya yang dipancarkan antenna lain (yang biasanya sudah distandarkan) pada daya pesawat pemancar radio yang sama. Contoh antenna standar adalah antenna isotropis, yang memancarkan sinyal secara merata ke segala arah berbentuk bola. Perbandingan daya dalam logaritmik:

dBm adalah nilai  $10 \log$  dari sinyal untuk 1 milli Watt

dBW adalah nilai  $10 \log$  dari sinyal untuk 1 Watt

Sinyal 100 milli Watt jika dijadikan dBm akan menjadi:

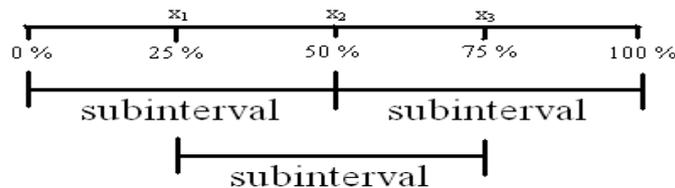
### GPS (Global Positioning System)

*Global positioning system* (GPS) adalah suatu sistem navigasi yang memanfaatkan satelit. Penerima GPS memperoleh sinyal dari beberapa satelit yang mengorbit bumi. Satelit yang mengitari bumi pada orbit pendek ini terdiri dari 24 susunan satelit, dengan 21 satelit aktif dan 3 buah satelit sebagai cadangan. Dengan susunan orbit tertentu, satelit GPS bisa diterima di seluruh permukaan bumi dengan penampakan antara 4 sampai 8 buah satelit. GPS dapat memberikan informasi posisi dan waktu dengan ketelitian sangat tinggi. Nama lengkapnya adalah NAVSTAR GPS (Navigational Satellite Timing and Ranging Global Positioning System); ada juga yang mengartikan "Navigation System Using Timing and Ranging." Dari perbedaan singkatan itu, orang lebih mengenal cukup dengan nama GPS. GPS mulai diaktifkan untuk umum 17 Juli 1995.

### Three-Point Interval Search

Metode ini digunakan untuk mendukung metode *Fletcher-Powell*. Tujuan dari metode ini adalah mencari nilai yang paling optimal di antara interval-interval yang ada, kemudian diiterasi ulang sampai didapatkan hasil yang kurang dari toleransi.

Dari interval yang diberikan dibagi menjadi 4 buah bagian dan fungsi objektif dievaluasi di 3 buah nilai interior. Dari fungsi ketiga nilai itu diambil yang terbaik, dimana jawabannya paling mendekati nilai yang diinginkan (maksimum atau minimum). Sub interval dari nilai yang dipilih akan diambil sebagai interval baru dan proses kembali diulang sampai selisih nilai fungsi yang didapatkan dengan yang sebelumnya kurang dari batas nilai toleransi. *Three-Point Interval Search* (Gambar 4) adalah prosedur paling efisien dalam pencarian yang menggunakan evaluasi interval ruang. Metode ini juga salah satu metode termudah untuk pencarian secara berurutan dalam kode untuk komputer.



Gambar 4. Three-point interval search.

## Metode Fletcher-Powell

Metode ini terdiri dari delapan langkah algoritma, dimulai dengan menentukan vektor  $\hat{X}$ , nilai toleransi  $\varepsilon$  dan matrix G sesuai dengan matrix identitas  $n \times n$ .  $\hat{X}$  dan G akan terus diperbaharui dengan perulangan iterasi sampai hasil yang didapatkan sama dengan atau kurang dari nilai  $\varepsilon$  dan hasil terakhir dari  $\hat{X}$  diambil sebagai hasil penyelesaian  $X^*$ .

Secara garis besar 8 langkah algoritma ini dapat dituliskan sebagai berikut:

- (1) Tentukan  $\hat{X}$  dan evaluasi  $\alpha = f(\hat{X})$  dan  $B = \nabla(f|\hat{X})$
- (2) Tentukan  $\lambda^*$  dimana  $f(\hat{X} + \lambda GB)$  adalah maksimum untuk  $\lambda = \lambda^*$ . Lalu set  $D = \lambda^* GB$
- (3) Perbaharui  $\hat{X}$  dengan  $\hat{X} + D$ .
- (4) Hitung  $\beta = f(\hat{X})$ . Bila  $\beta - \alpha \leq \varepsilon$  maka lanjutkan ke langkah 5, jika tidak lanjut ke langkah 6.
- (5) Set  $X^* = \hat{X}$  dan  $f(X^*) = \beta$  dan stop
- (6) Evaluasi  $C = \nabla(f|\hat{X})$  dan set  $Y = B - C$ .
- (7) Kalkulasi matrix  $n \times n$

$$L = \left( \frac{1}{D^T Y} \right) D D^T \quad M = \left( \frac{-1}{Y^T G Y} \right) G Y Y^T G$$

- (8) Perbaharui nilai G dengan  $G + L + M$ . Perbaharui  $\alpha$  dengan nilai dari  $\beta$ . Perbaharui B dengan nilai dari C, dan kembali ke langkah 2. (Bronson & Naadimuthu, 1997).

## METODE

Untuk membantu perancangan proses maka akan dirancang dahulu diagram UML untuk membantu proses pembuatan aplikasi. Diagram UML yang akan dibuat adalah *activity diagram* (Pressman RS, 2002). Pertama-tama *user* akan menjalankan aplikasi, lalu sistem akan menampilkan menu utama. Di menu utama akan muncul dua pilihan yaitu 'Jalankan Program' dan 'Help'. Bila *user* memilih 'Help', akan tampil menu *Help* yang akan memberikan penjelasan mengenai cara penggunaan program aplikasi, lalu kemudian *user* akan diberikan kembali dua pilihan yaitu 'Jalankan

Program' atau pilih 'Exit' untuk keluar dari program aplikasi. Jika *user* memilih 'Jalankan Program', *user* akan diminta untuk memasukkan jumlah data, lalu masukkan semua input data, setelah itu sistem akan menampilkan semua data yang sudah dimasukkan. *User* diminta untuk memeriksa kembali semua data yang telah dimasukkan, apabila tidak benar, tekan tombol T untuk kembali memasukkan jumlah data beserta input datanya. Apabila data yang dimasukkan sudah benar tekan tombol Y. Setelah itu *user* akan ditanya apakah iterasi yang akan dilakukan akan diperlihatkan. Apabila tidak, tekan tombol T. Sistem akan langsung memperlihatkan hasil akhir. Apabila ya, tekan tombol Y untuk memulai iterasi yang pertama. Sistem akan langsung memulai iterasinya yang pertama dan menunjukkan hasil perhitungan iterasinya ke *user*. Apabila hasilnya belum optimal, *user* dapat menekan sembarang tombol untuk melanjutkan ke iterasi berikutnya sampai hasil perhitungan iterasi sudah optimal. Berikut rancangan proses yang telah disusun berdasarkan aplikasi di atas (Gambar 5).

### Perancangan Teknis Optimasi Jarak

Langkah-langkah yang dijalankan dalam menghitung pengoptimasian dengan menggunakan algoritma *Fletcher-Powell* adalah:

- (1) *user* memasukkan koordinat-koordinat *Main Coverage Point* yang diinginkan. Untuk mencegah terjadinya kesalahan perhitungan, *user* hanya diperbolehkan memasukkan 10 buah koordinat *Main Coverage Point*;
- (2) menentukan posisi koordinat sementara (Gambar 6), fungsi objektif, gradien vektor, beserta inisialisasi toleransi.

$(x_s, y_s)$  adalah koordinat sementara penempatan pemancar yang didapat dengan cara mengambil nilai rata-rata dari semua koordinat-koordinat *Main Coverage Point*. Jadi setiap koordinat *Main Coverage Point* dijumlahkan seluruhnya dan hasilnya dibagi dengan jumlah *Main Coverage Point* ( $n$ ) yang ada.

$$(x_s, y_s) = (\bar{x}, \bar{y}) = \begin{bmatrix} x_s \\ y_s \end{bmatrix} = \frac{\begin{bmatrix} x_1 + x_2 + \dots + x_n \\ y_1 + y_2 + \dots + y_n \end{bmatrix}}{n}$$

$\alpha$  adalah fungsi objektif minimum yang didapat dengan cara menjumlahkan semua jarak antara tiap koordinat *Main Coverage Point* dengan koordinat posisi pemancar sementara.

$$\alpha = r_1 + r_2 + \dots + r_n$$

Karena metode ini biasanya digunakan untuk mencari kondisi maksimum, fungsi objektif dikalikan dengan bilangan -1.

$$\alpha = f \begin{bmatrix} x_s \\ y_s \end{bmatrix} = -\sqrt{(x_s - x_1)^2 + (y_s - y_1)^2} - \sqrt{(x_s - x_2)^2 + (y_s - y_2)^2} - \dots - \sqrt{(x_s - x_n)^2 + (y_s - y_n)^2} \quad \nabla f =$$

gradien vektor koordinat pemancar terhadap tiap-tiap koordinat *Main Coverage Point*.

$$\nabla f \begin{bmatrix} x_s \\ y_s \end{bmatrix} = \begin{bmatrix} -\frac{x_s - x_1}{\sqrt{(x_s - x_1)^2 + (y_s - y_1)^2}} - \frac{x_s - x_2}{\sqrt{(x_s - x_2)^2 + (y_s - y_2)^2}} - \dots - \frac{x_s - x_n}{\sqrt{(x_s - x_n)^2 + (y_s - y_n)^2}} \\ -\frac{y_s - y_1}{\sqrt{(x_s - x_1)^2 + (y_s - y_1)^2}} - \frac{y_s - y_2}{\sqrt{(x_s - x_2)^2 + (y_s - y_2)^2}} - \dots - \frac{y_s - y_n}{\sqrt{(x_s - x_n)^2 + (y_s - y_n)^2}} \end{bmatrix} = \begin{bmatrix} Vx \\ Vy \end{bmatrix}$$

$\varepsilon$  adalah sebuah nilai toleransi yang digunakan untuk mengukur ketelitian hasil perhitungan. Semakin kecil nilai toleransi, maka hasil perhitungan akan semakin akurat. Inisialisasi dari  $\varepsilon$  untuk program ini adalah 0.005 (Page, 1980).

(3) inialisasi koefisien matriks pengali gradien vektor dan menentukan nilai koefisien skalar pengali gradien vektor dengan metode *three point interval search*.

G adalah koefisien matriks pengali gradien vektor. Inialisasi awal G sebuah matriks identitas.

$$G = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$\lambda$  adalah koefisien skalar pengali gradien vektor, didapat dengan cara mengoptimalkan fungsi  $\alpha$  setelah koordinat pemancar sementara dijumlahkan dengan  $\lambda \cdot G \cdot \nabla f$ .

$\lambda$  diambil pada saat  $f\left(\begin{bmatrix} x_s \\ y_s \end{bmatrix} + \lambda G \begin{bmatrix} Vx \\ Vy \end{bmatrix}\right)$  mencapai maksimum

(4) bandingkan selisih  $f\left(\begin{bmatrix} x_s \\ y_s \end{bmatrix}\right)$  dengan  $f\left(\begin{bmatrix} x_s \\ y_s \end{bmatrix} + \lambda G \begin{bmatrix} Vx \\ Vy \end{bmatrix}\right)$ .

Apabila kurang dari toleransi, langsung didapatkan koordinat hasil optimasi yaitu  $\begin{bmatrix} x_s + \lambda G Vx \\ y_s + \lambda G Vy \end{bmatrix}$  dan berhenti disini.

(5) apabila selisih antara  $f\left(\begin{bmatrix} x_s \\ y_s \end{bmatrix}\right)$  dengan  $f\left(\begin{bmatrix} x_s \\ y_s \end{bmatrix} + \lambda G \begin{bmatrix} Vx \\ Vy \end{bmatrix}\right)$  masih lebih besar dari toleransi, proses optimasi akan diulang dengan meng-*update* nilai dari  $\alpha$ ,  $\begin{bmatrix} x_s \\ y_s \end{bmatrix}$ , G,  $\nabla f\left(\begin{bmatrix} x_s \\ y_s \end{bmatrix}\right)$ .

(6)  $\alpha$  akan diupdate menjadi  $f\left(\begin{bmatrix} x_s \\ y_s \end{bmatrix} + \lambda G \begin{bmatrix} Vx \\ Vy \end{bmatrix}\right)$ .

(7)  $\begin{bmatrix} x_s \\ y_s \end{bmatrix}$  akan diupdate menjadi  $\begin{bmatrix} x_s \\ y_s \end{bmatrix} + \lambda G \begin{bmatrix} Vx \\ Vy \end{bmatrix}$ .

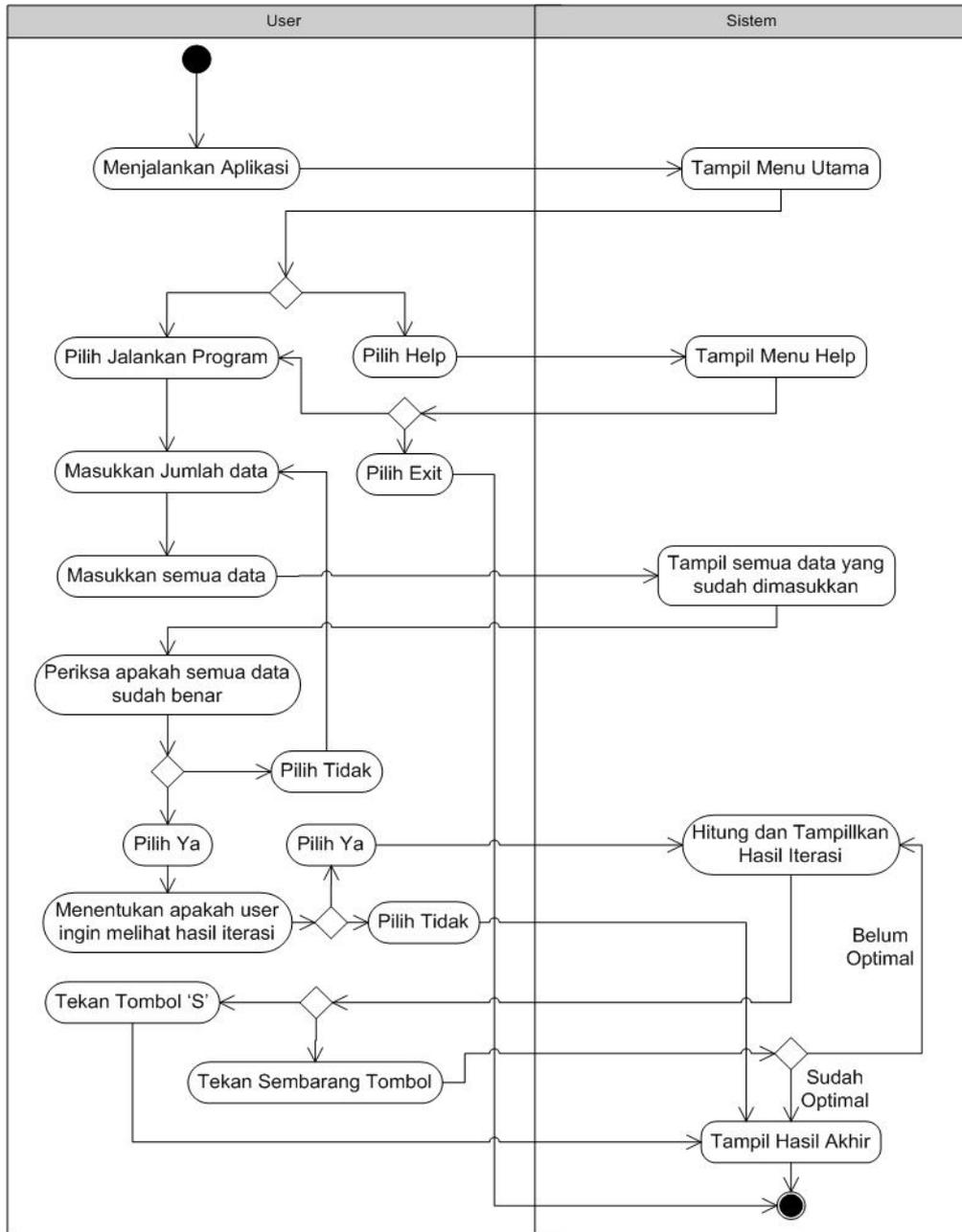
(8)  $D = \lambda GB$  dan  $Y = \nabla f\left(\begin{bmatrix} x_s \\ y_s \end{bmatrix}\right) - \nabla f\left(\begin{bmatrix} x_s \\ y_s \end{bmatrix} + \lambda G \begin{bmatrix} Vx \\ Vy \end{bmatrix}\right)$ , maka

G akan di-*update* menjadi  $G + \left(\frac{1}{D^T Y}\right) D D^T + \left(\frac{-1}{Y^T G Y}\right) G Y Y^T G$ .

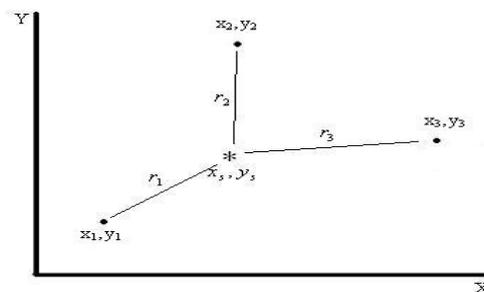
(9)  $\nabla f\left(\begin{bmatrix} x_s \\ y_s \end{bmatrix}\right)$  akan di-*update* menjadi  $\nabla f\left(\begin{bmatrix} x_s \\ y_s \end{bmatrix} + \lambda G \begin{bmatrix} Vx \\ Vy \end{bmatrix}\right)$ .

(10) lalu kembali mencari nilai  $\lambda$  pada saat  $f\left(\begin{bmatrix} x_s \\ y_s \end{bmatrix} + \lambda G \begin{bmatrix} Vx \\ Vy \end{bmatrix}\right)$  mencapai maksimum.

(11) proses diulang sampai selisih  $f\left(\begin{bmatrix} x_s \\ y_s \end{bmatrix}\right)$  dengan  $f\left(\begin{bmatrix} x_s \\ y_s \end{bmatrix} + \lambda G \begin{bmatrix} Vx \\ Vy \end{bmatrix}\right)$  kurang dari toleransi.



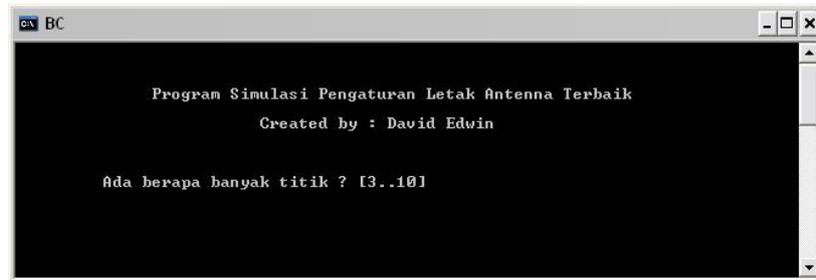
Gambar 5. Activity Diagram dari rancangan program aplikasi yang akan dibuat.



Gambar 6. Posisi koordinat pemancar hasil optimasi sementara.

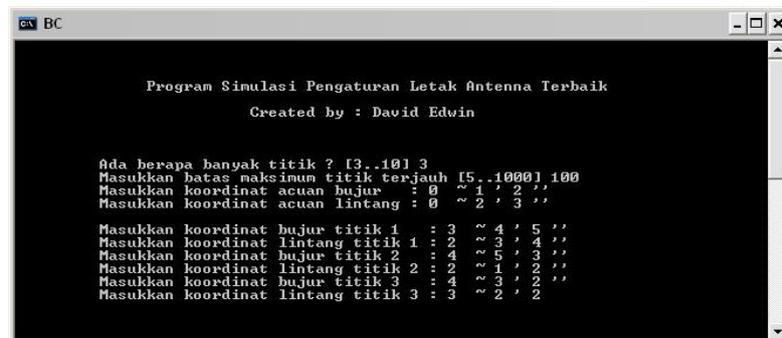
## HASIL DAN PEMBAHASAN

Setelah memperhatikan langkah-langkah perancangan sebelumnya, telah dibuat program komputer untuk proses perhitungannya dan dihasilkan beberapa tampilan seperti yang ditunjukkan berikut ini: Setelah *user* memilih nomor 1 pada layar menu utama, akan dijumpai tampilan layar seperti Gambar 7.



Gambar 7. Tampilan layar menu 'Jalankan Program'.

Pada layar ini *user* diminta untuk memasukkan jumlah koordinat target yang diinginkan. Batas jumlah koordinat yang dimasukkan tergantung pada variasi angka-angka yang akan dimasukkan. Karena angka yang akan dimasukkan tidak dapat diduga, pada program aplikasi ini banyaknya titik akan dibatasi sampai 10 titik saja. Setelah memasukkan jumlah koordinat yang diinginkan (Gambar 8), *user* diminta untuk memasukkan batas maksimum titik terjauh. Nilai yang akan dimasukkan akan dibatasi dari 5 sampai dengan 1000.



Gambar 8. Tampilan layar menu ketika data sedang dimasukkan.

Setelah *user* memasukkan semua input yang diinginkan, *user* dapat memeriksa kembali data-data yang sudah dimasukkan. Apabila data yang dimasukkan ada yang tidak benar, *user* dapat memilih 'T' untuk memasukkan ulang semua data-data. Bila data yang dimasukkan sudah benar, *user* dapat memilih 'Y' dan melanjutkan ke langkah berikutnya.

Langkah berikutnya *user* dapat kembali memilih untuk melihat hasil iterasi atau tidak. Bila ya, *user* dapat memilih 'Y'. Bila tidak, *user* dapat memilih 'T' dan kembali melanjutkan ke langkah berikutnya. Sistem akan mulai mencari nilai  $\beta$  dan dibandingkan dengan nilai  $\alpha$  sampai selisihnya kurang dari nilai toleransi.

Iterasi akan terus dilakukan sampai selisih hasil yang didapat dengan hasil sebelumnya kurang dari toleransi. Bila *user* menekan tombol 'S', iterasi-iterasi yang ada tidak akan diperlihatkan dan langsung menunjukkan hasil akhir (Gambar 9).

```

BC
-----
Koordinat acuan bujur 0° 1' 2'' BT
Koordinat acuan lintang 0° 2' 3'' LU
Titik 1 : 3° 4' 5'' BB 8 2° 3' 4'' LS ----> <-342,-231>
Titik 2 : 4° 5' 3'' BB 8 2° 1' 2'' LU ----> <-455,-220>
Titik 3 : 4° 3' 2'' BT 8 3° 2' 2'' LS ----> <417,-340>

JAWABAN AKHIR
Selisih akhir = -0.000977          Banyak Iterasi = 8
Koordinat W = -311.112579        Variansi W = 484131.062500
Koordinat V = -195.222473        Variansi V = 176725.312500
Jarak Total = 1260.160400        Variansi Total = 660856.375000
                                   Waktu awal : 11:05:49.75
                                   Lama Proses : 0:02:17.09

Jarak terjauh yang diperbolehkan = 100
Jarak Terjauh = 772.603021
Luas minimum jangkauan Antenna = 1874318.324696
Koordinat bujur = 2° 47' 8'' BB
Koordinat lintang = 1° 43' 29'' LS

Titik 2 Berada di luar Jangkauan pemancar
Titik 3 Berada di luar Jangkauan pemancar

```

Gambar 9. Tampilan layar 'Jawaban akhir'.

Proses pengujian program dilakukan dengan maksud untuk mengetahui kelebihan dan kekurangan apa saja yang dimiliki oleh program ini. Pengujian dilakukan dengan memasukkan data acak ke dalam program aplikasi. Hasil pengujian dapat dilihat pada Tabel 1 di bawah ini.

Tabel 1  
Hasil pengujian dari beberapa data secara acak

Banyak Data	Banyak Iterasi	Variansi Data	Lama Proses (detik)
3	2	931	0,05
3	7	2500	0,26
3	2	4683	0,06
4	8	4282	0,22
4	9	11413	0,27
4	4	2769	0,11
5	6	7894	0,16
5	5	7941	0,16
5	3	7078	0,11
6	2	8948	0,06
6	6	5638	0,17
6	5	6386	0,17
7	7	10868	0,22
7	6	10637	0,17
7	5	8699	0,11
8	5	10539	0,16
8	2	10655	0,06
8	9	12074	0,22
9	8	7730	0,21
9	6	17234	0,11
9	2	13377	0,06
10	6	16620	0,16
10	3	16730	0,05
10	4	13942	0,11

Dari hasil-hasil pengujian diperoleh bahwa banyaknya iterasi tidak dapat ditentukan dengan banyaknya data yang dimasukkan, lama proses perhitungan relatif bergantung terhadap banyaknya iterasi, semakin banyak iterasi yang dilakukan, waktu proses yang dibutuhkan relatif lebih besar, banyak data relatif akan memperbesar nilai variansi dan variansi tidak mempengaruhi waktu proses yang dibutuhkan.

Setelah melakukan serangkaian pengujian terhadap program aplikasi, didapatkan beberapa kelebihan dan kekurangan program aplikasi. Adapun kelebihan adalah proses perhitungan yang dilakukan program aplikasi ini relatif cepat, mudah untuk dimengerti dan digunakan (*user friendly*). Program ini berbasis DOS sehingga tidak menggunakan ruang *hard disk* yang besar dan penggunaan *memory* yang relatif kecil. Kelemahan program ini mempunyai jangkauan angka yang relatif kecil, sehingga memungkinkan terjadinya *error* apabila nilai yang dihasilkan berada di luar jangkauan serta penampilan *user interface* yang kurang menarik.

## PENUTUP

Berdasarkan pada pembahasan sebelumnya dan hasil pengujian terhadap aplikasi optimasi jarak menggunakan Algoritma *Fletcher-Powell* yang telah berhasil dirancang dapat disimpulkan bahwa Algoritma *Fletcher-Powell* dapat digunakan untuk mencari optimasi dari suatu fungsi jarak yang menggunakan posisi koordinat Cartesian dan algoritma *Fletcher-Powell* yang dirancang hanya dapat digunakan untuk mencari optimasi minimum. Untuk pencarian optimasi yang maksimum, algoritma ini tetap dapat digunakan dengan mengkalikan fungsi objektifnya dengan bilangan negatif.

Dalam kesempatan ini penulis mengucapkan banyak terima kasih kepada David Edwin yang telah membantu dalam pembuatan program simulasi untuk perhitungan dan tampilan layar penelitian ini.

## DAFTAR PUSTAKA

- Bronson, R., Naadimuthu, G. (1997). *Operations Research*, (2<sup>nd</sup> ed.). New York: McGraw Hill.
- Gstatic.com. (n.d.). Diakses dari [http://t0.gstatic.com/images?q=tbn:ANd9GcQjNzh6xDI\\_g-SIB18zJ2Y2KFwSdZPoYDPiNSQCqyDzmFpspqy7KQ](http://t0.gstatic.com/images?q=tbn:ANd9GcQjNzh6xDI_g-SIB18zJ2Y2KFwSdZPoYDPiNSQCqyDzmFpspqy7KQ).
- Kanginan, Marthen. (2000). *Fisika 2000 Jilid 2A & 3A untuk SMU Kelas 2*. Jakarta: Penerbit Erlangga.
- Page, Jones M. (1980). *The Practical Guide to Structural System Design*. New York: Yourdon Press.
- Pressman, R.S. (2002). *Rekayasa Perangkat Lunak Pendekatan Praktis*, (Buku Satu). (C.N.Harnaningrum, terj.). Yogyakarta: Penerbit Andi.
- Teknikcall-yogya.com. (n.d.). <http://teknikcall-yogya.com/wp-content/uploads/2011/08/Pola-radiasi-pancaran-pada-Antena-Parabolic-300x123.png&w=300&h=123&ei=tUHTTqy5K86ziQeh1qSyDg&zoom=1>