IMPLEMENTASI ALGORITMA GENETIKA DAN METODE VAM UNTUK MENYELESAIKAN MASALAH TRANSPORTASI LINIER

Dina Thaib[1], Nirwansyah APN[2], Suyanto[3]
UPBJJ-UT Bandung, Bandung
STT-Telkom, Bandung
STT-Telkom, Bandung

(dinathaib@ut.ac.id)

ABSTRAK

Model masalah transportasi linier, merupakan permasalahan program linier, pada umumnya berkaitan dengan penentuan biaya pengiriman terendah untuk mendistribusikan sembarang komoditi dari sembarang kelompok pemasok, yang disebut sebagai **sumber,** ke sembarang penerima, yang disebut sebagai **tujuan**. *Vogel Approximation Model* (VAM) yang dianggap sebagai metode konvesional terbaik yang dapat menyelesaikan masalah transportasi linier masih memiliki kelemahan. Solusi awal yang dihasilkan dari metode ini terkadang masih jauh dari solusi optimal. Algoritma Genetika merupakan metode pencarian yang handal yang memiliki kemampuan menemukan solusi yang optimal atau paling tidak mendekati optimal untuk masalah transportasi. Salah satu parameter yang dapat menentukan biaya pengiriman terendah pada masalah transportasi adalah jarak dari sumber ke tujuan. Algoritma A* merupakan salah satu algoritma pencarian yang digunakan untuk mencari jalur dan jarak terpendek dari sumber ke tujuan. Makalah ini akan membahas penerapan algoritma Genetika pada masalah transportasi linier dengan masukan jarak sebagai biaya yang dihasilkan melalui Algoritma A*. Setiap solusi awal yang dihasilkan melalui algoritma Genetika dibandingkan dengan metode VAM. Untuk ukuran percobaan mxn ≤ 25 di mana m titik penawaran pada sumber dan n titik permintaan pada tujuan, solusi awal yang diperoleh dengan menggunakan algoritma Genetika sama atau lebih baik dibandingkan dengan VAM pada waktu yang sama. Untuk mxn > 25, algoritma Genetika membutuhkan waktu yang lebih besar untuk menemukan solusi awal, yang mengikuti kurva persamaan kuadrat.

Kata kunci: masalah transportasi linier, VAM, algoritma A*, algoritma genetika

PENDAHULUAN

Masalah transportasi merupakan masalah yang sering dihadapi dalam pendistribusian barang yang tujuannya adalah bagaimana menentukan pendistribusian barang tersebut dari suatu sumber ke tujuan dengan biaya seminimum mungkin. Beberapa metode peneyelesaian konvensional dalam masalah transportasi yang biasa digunakan memiliki kekurangan dan kelebihan yang berbeda. Metoda barat laut misalnya, merupakan metoda yang paling mudah namun memiliki lebih banyak iiterasi untuk mencapai solusi yang optimal dibandingkan metoda biaya terendah ataupun metode pendekatan Vogel (Siang, J.J, 2011). Walaupun lebih baik dari metode barat laut, metode biaya terendah, atau yang sering disebut sebagai metode Greedy cenderung menghasilkan solusi awal yang tidak optimum. Metode pendekatan Vogel yang dianggap sebagai metode konvensional yang terbaik, juga tidak selalu dapat menemukan solusi awal yang optimum (Taha, H.A,1992 dan Siang, J.J, 2011).

Algoritma genetika yang diinspirasi dari teori evolusi Darwin yang selanjutnya diadopsi menjadi algoritma komputasi untuk mencari solusi suatu permasalahan dengan cara yang lebih alami, memiliki potensi sebagai teknik optimasi untuk masalah-masalah yang kompleks, salah satunya adalah masalah transportasi.

Menemukan biaya terendah pada masalah transportasi pada dasarnya adalah mencari jarak terdekat antara satu simpul dengan simpul lain. Algoritma A* merupakan algoritma pencarian yang memberikan solusi yang cukup manjur bagi proses pencarian jalan. Pada makalah ini selanjutnya akan dibahas bagaimana algoritma genetika dapat memecahkan masalah transportasi liner dengan masukan jarak/biaya yang dihasilkan melalui algoritma A* yang digunakan untuk mencari jarak terdekat antara sumber dan tujuan.

METODOLOGI

Model Transportasi

Model dari suatu permasalahan transportasi dari suatu jaringan dengan **m** sumber dan **n** tujuan, di mana masing-masing sumber dan tujuan diwakili oleh sebuah simpul serta busur yang menghubungkan sebuah sumber dan sebuah tujuan, mewakili jalur pengiriman suatu komiditas tertentu, diberikan oleh persamaan berikut ini :

$$\text{Minimumkan} : x = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \qquad (2.1)$$

Dengan batasan

$$\sum_{j=1}^{n} x_{ij} \leq a_i \qquad i = 1,2,.....m \qquad (2.2)$$

$$\sum_{i=1}^{m} x_{ij} \leq b_j \qquad j = 1,2,.....n \qquad (2.3)$$

$$x_{ij} = 0, \qquad \text{untuk semua } i \text{ dan } j \qquad (2.4)$$

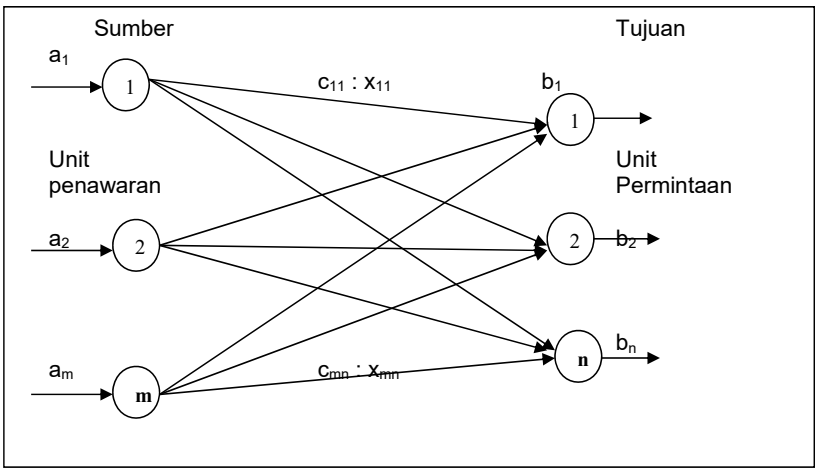
Dimana :.

a_i = jumlah penawaran di sumber i

b_j = jumlah permintaan dari tujuan j

c_{ij} = biaya unit transportasi antara sumber i dan tujuan j

x_{ij} = jumlah barang yang dikirim dari sumber i ke tujuan j



Model transportasi berimbang terjadi jika jumlah penawaran total = jumlah penerimaan total [Taha, 1992]

$$\sum_{j=1}^{n} x_{ij} \leq a_i = \sum_{i=1}^{m} x_{ij} \leq b_j$$

Pertidaksamaan di atas menjadi:

$$\sum_{j=1}^{n} x_{ij} = a_i \qquad i = 1,2,.....m \qquad (2.5)$$

$$\sum_{i=1}^{m} x_{ij} = b_j \qquad j = 1,2,.....n \qquad (2.6)$$

Diperoleh:

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j \qquad (2.7))$$

Solusi layak dari masalah transportasi akan dapat diperoleh apabila asumsi kondisi selalu berimbang,.Solusi optimal akan dimiliki oleh suatu masalah programa linier dengan batasan tertentu yang memiliki solusi layak.(Bazaraa et.al, 1990)
Model transportasi dapat diringkas dalam bentuk tabel transportasi berikut ini :

Tujuan

Sumber: ke / dari	1	2	.....	n	Persediaan
1	x_{11} c_{11}	x_{12} c_{12}		x_{1n} c_{1n}	a_1
2	x_{21} c_{21}	x_{22} c_{22}		x_{2n} c_{2n}	a_2
.....	.....	.....		.....	.....
m	x_{m1} c_{m1}	x_{m2} c_{m2}		x_{mn} c_{mn}	a_m
Permintaan	b_1	b_2	…….	b_n	

Algoritma Pencarian

Merupakan suatu algoritma untuk menemukan solusi lintasan melalui ruang masalah yang diberikan. Karena lintasan mengandung suatu rangkaian operasi yang dilakukan sehingga ditemukan solusi masalah, maka algoritma ini harus dapat mencari lintasan dari simpul asal hingga simpul tujuan. Salah satu masalah yang timbul dalam perancangan algoritma pencarian pada graf adalah solusi dapat dicapai melalui lintasan yang berbeda. Oleh karena itu algoritma ini harus mampu mendeteksi dan menghilangkan putaran (*loop*) dari solusi lintasan yang diberikan. Ada dua jenis algoritma pencarian, tanpa panduan dan dengan panduan. Algoritma pencarian tanpa panduan adalah algoritma yang pemilihan simpulnya tidak mempertimbangkan biaya dari lintasan yang dilalui. Sedangkan algoritma pencarian dengan panduan adalah algoritma yang pemilihan simpulnya mempertimbangkan pemilihan lintasan yang dilalui berdasarkan panduan yang diberikan, disebut sebagai *heuristic,* yang merupakan . suatu teknik yang meningkatkan efisiensi proses pencarian. Dengan menggunakan teknik ini, solusi yang baik dari masalah yang kompleks dapat ditemukan, walaupun mungkin solusi tersebut tidak optimal.

Fungsi *heuristic* adalah suatu fungsi yang bertujuan untuk memandu pencarian pada arah yang paling menguntungkan dengan menyarankan lintasan mana yang dilalui pertama kali dari banyak lintasan yang tersedia (Rich, E,1990).

Salah satu algoritma pencarian dengan panduan adalah algoritma A* yang merupakan penyederhanaan dari algoritma *Best First Search* (BFS).

Best First Search

Merupakan salah satu teknik pencarian jalur terpendek secara *heuristic* yang mengkombinasikan kelebihan dari teknik *Dept First Search* (DFS) dan BFS. Keuntungan DFS adalah solusi dapat ditemukan tanpa semua simpul harus dikembangkan, sedangkan keuntungan BFS adalah tidak akan terjebak pada lintasan yang buntu. Kedua teknik ini dikombinasikan dengan menjalani lintasan tertentu tetapi berganti ke lintasan lainnya apabila lintasan lainnya memiliki nilai evaluasi lebih baik dari lintasan yang dijalaninya saat ini. Dalam prakteknya, BFS tidak efisien.

Algoritma A*

Prinsip dasar algoritma ini adalah mencari jalur terpendek dari suatu simpul awal menuju simpul tujuan dengan memperhatikan biaya terendah. Algoritma A* menyelesaikan masalah yang menggunakan graf untuk perluasan ruang statusnya. Melalui penerapan suatu heuristic, langkah-langkah yang tidak perlu dibuang dengan pertimbangan bahwa langkah-langkah tersebut merupakan langkah yang tidak akan mencapai solusi yang diinginkan. Algoritma A* membangkitkan simpul yang paling mendekati solusi, kemudian disimpan suksesornya ke dalam *list* sesuai dengan urutan yang paling mendekati solusi terbaik. Langkah selanjutnya, simpul pertama pada *list* diambil, dibangkitkan suksesornya dan kemudian suksesor ini disimpan ke dalam *list* sesuai dengan urutan yang terbaik untuk solusi. Untuk membantu penyelesaian persoalan, jarak estimasi/heuristic (h(n)) dan jarak sesungguhnya/*cost* (g(n)) digabungkan. *Heuristic* adalah nilai yang memberi harga pada tiap simpul yang memandu A* mendapatkan solusi yang diinginkan. *Heuristic* adalah fungsi optimasi yang menjadikan algoritma A* lebih baik dari pada algoritma lainnya. Namun *heuristic* masih merupakan estimasi/perkiraan biasa saja, tidak ada rumus khususya. . Artinya, setiap kasus memiliki fungsi heuristic yang berbeda-beda. (http://adit279.wordpress.com/2008/12/04/implementasi-algoritma-a-pada-permasalahan-optimasi-solusi-dynamic-water-jug/)

Fungsi evaluasi :

$$f(n) = g(n) + h(n)$$

dimana :

n = simpul yang dilalui dalam graf

g(n) = biaya dari simpul asal menuju simpul n

h(n) = estimasi heuristic biaya dari simpul n menuju simpul tujuan

Apabila fungsi evaluasi digunakan pada BFS, hasilnya disebut sebagai algoritma A Misalkan

$$f^*(n) = g^*(n) + h^*(n)$$

dimana :

n = simpul yang dilalui dalam graf

$g^*(n)$ = biaya jalur terpendek dari simpul n menuju simpul tujuan

$h^*(n)$ = biaya sebenarnya untuk jalur terpendek dari simpul n menuju simpul tujuan

Apabila f* digunakan pada BFS, hasil dari strategi pencarian dapat diterima. Pada kebanyakan permasalahan nyata diupayakan f merupakan pendekatan f*. Jika algoritma A menggunakan fungsi evaluasi f dimana $h(n) \leq h^*(n)$, disebut algoritma A*. Dengan demikian dapat dikatakan algoritma A* dapat diterima

Algoritma A* dipilih sebagai algoritma pencarian jalur terpendek berdasarkan beberapa pertimbangan sebagai berikut (Nirwansyah, 2002) :

1. Merupakan pengembangan dari BFS yang merupakan algoritma jalur terpendek yang dapat diterima
2. Representasi ruang *state* pada BFS sendiri adalah graf berakar yang memiliki karakteristik yang bersesuaian dengan bentuk permasalahan yang ada.
3. Penggunaan fungsi heuristic pada algoritma A* menjamin bahwa jalur yang ditemukan selalu jalur terpendek dan memungkinkan untuk diimplementasikan.
4. Memperhitungkan dua buah fungsi dalam menentukan *state* berikutnya yakni fungsi **g(n)** yang merupakan fungsi biaya dan **h(n)** yang merupakan fungsi heuristic yang memandu pemilihan *state* berikutnya

Sifat dari algoritma A* pada masalah transportasi adalah :

1. n adalah setiap simpul yang dilalui dalam graf untuk setiap simpul sumber ke setiap simpul tujuan
2. g(n) adalah biaya dari simpul sumber menuju simpul n
3. h(n) adalah perkiraan heuristic biaya dari simpul n menuju simpul tujuan yaitu jarak garis lurus simpul n ke simpul tujuan sehingga dapat dijamin bahwa $h(n) \leq h^*(n)$

Algoritma Genetika

Merupakan teknik pencarian yang didasarkan atas mekanisme seleksi dan genetika alamiah (Gen, M & R. Cheng, 1997) dan pada dasarnya merupakan program komputer yang mensimulasikan proses evolusi. Populasi dari kromosom dihasilkan secara acak dan memungkinkan untuk berkembang biak sesuai dengan hukum evolusi dengan harapan akan memperoleh individu kromosom yang prima. Kromosom merupakan kandidat solusi permasalahan yang bila berkembang dengan baik diharapkan akan menghasilkan solusi yang baik pula. Sebuah kromosom merupakan rangkaian simbol berupa rangkaian bit biner dan berkembang melalui iterasi berturut-turut yang disebut sebagai generasi. Pada setiap generasi, berdasarkan tingkat *fitness*, sertiap kromosom dievaluasi. Kromosom baru (*offspring*) yang merupakan generasi berikutnya, dibentuk melalui dua cara :

1. menggunakan operator *crossover* untuk menggabungkan dua kromosom dari generasi saat ini
2. menggunakan operator *mutation* untuk memodifikasi sebuah kromosom

Generasi yang baru dibentuk dengan cara :

1. *selecting*, berdasarkan nilai *fitness* dari *parent* dan *offspring*
2. menolak generasi baru untuk menjaga populasi tetap

Crossover merupakan operator genetik utama yang mengoperasikan dua kromosom sekaligus pada saat yang bersamaan dan menghasilkan *offspring* dengan mengkombinasikan kelebihan-kelebihan pada kedua kromosom. Cara sederhana untuk melakukan *crossover* adalah dengan memilih *parent* secara acak kemudian membangkitkan *offspring* dengan mengkombinasikan bagian kiri salah satu parent dengan bagian kanan dari *parent* lainnya. Metode ini bekerja dengan sangat baik pada representasi rangkaian bit. Kinerja algoritma genetika sangat tergantung pada kinerja operator *crossover* yang digunakan.

Mutation merupakan operator yang secara spontan memproduksi perubahan acak pada kromosom berbeda dan cara termudah untuk memperoleh keberhasilan *mutation* adalah dengan mengubah satu atau lebih gen. Pada algoritma genetika, *mutation* memegang peranan penting yaitu:

1. mengganti gen-gen yang hilang dari populasi melalui proses seleksi sehingga dapat dilakukan percobaan lainnya
2. menyediakan gen-gen yang tidak terdapat pada populasi awal.

Algoritma genetika berbeda dengan prosedur optimasi konvensional dan pencarian dalam beberapa hal mendasar (Nirwansyah, 2002):

1. algoritma genetika bekerja pada himpunan solusi, bukan pada suatu solusi
2. algoritma genetika mencari dari kumpulan solusi, bukan dari solusi tunggal

3. algoritma genetika menggunakan informasi *payoff* (fungsi *fitness*), bukan turunan atau pengetahuan lainnya
4. algoritma genetika menggunakan aturan transisi probabilistik bukan aturan deterministik

Secara singkat langkah-langkah dari Algoritma genetika klasik diberikan sebagai berikut.

1. *Start*: membentuk secara acark populasi n buah kromosom (solusi layak untuk permasalahan)
2. *Fitness* : melakukan evaluasi kecocokkan f(x) dari setiap kromosom x pada populasi
3. Populasi baru : bentuk populasi baru dengan mengulang langkah-langkah berikut ini hingga populasi baru lengkap.
   a. *Selection* : Pilih dua *parent* kromosom dari suatu populasi berdasarkan kecocokkan mereka (kecocokan yang lebih baik menghasilkan peluang lebih besar untuk dipilih)
   b. *Crossover* : dengan suatu persilangan memungkinkan persilangan *parent* untuk membentuk suatu *offspring* baru. Jika persilangan tidak terjadi, maka *offspring* yang terbentuk adalah identik dengan *parent*.
   c. *Mutation* : Dengan suatu muatsi kemungkinan mutasi *offspring* baru pada setiap lokus (posisi pada kromosom)
   d. *Accepting* : Tempatkan *offspring* baru pada suatu populasi baru
4. *Replace*. Gunakan populasi yang baru terbentuk untuk menjalankan algoritma selanjutnya
5. Evaluasi : Apabila kondisi akhir memuaskan, *stop,* dan kembalikan solusi terbaik dalam populasi saat ini
6. *Loop* : Menuju langkah 2.

HASIL DAN PEMBAHASAN

Perancangan dan implementasi sistem

Perancangan mencakup proses penerimaan masukan dari pengguna, proses pencarian jarak terpendek dari simpul sumber ke simpul tujuan, proses pencarian ruang solusi untuk distribusi hasil distribusi dengan menggunakan algoritma genetika dan proses evaluasi dari ruang solusi yang diberikan. Langkah-langkah pada tahap perancangan mencakup penyusunan Diagram Aliran Data (DAD), kamus data dan deskripsi proses. DAD tersusun dalam tiga level yakni :

1. level 1 mencakup proses pencarian jalur terpendek, pencarian ruang solusi dan evaluasi ruang solusi
2. level 2 merupakan rincian proses pencariang ruang solusi yang mencakup pencarian ruang *parents* dan penurunan ruang *offspring*
3. level 3 merupakan rincian proses penurunan ruang *offspring* yang mencakup proses *crossover* dan *mutation*

Kamus data merupakan catatan tentang nama, deskripsi, tipe dan struktur data dari masing-masing proses yang tersusun pada DAD. Deskripsi proses mencakup nama, deskripsi dan logika proses dari masing-masing proses yang tersususn pada DAD.

Implementasi sistem mencakup: 1) spesifikasi umum perangkat lunak yang digunakan serta lingkup perangkat keras yang digunakan dalam pengembangan dan pengujian sistem ; 2) algoritma pencarian jalur terpendek; dan 3) algoritma genetika pada masalah transportasi. Perangkat lunak yang digunakan pada pengembangan sistim ini adalah Microsoft Visual Basic 6 pada sistem operasi Microsoft Office 2000 Server. Aplikasi pendukung lain yang digunakan adalah MapInfo Professional 6. Sedangkan lingkungan perangkat keras yang digunakan dalam pengembangan dan pengujian adalah komputer dengan prosessorAMD Duron 650 MHz dengan memory 384 MB.

Analisa Sistem

Untuk jalur transportasi digunakan peta digital kota Bandung sebagai acuan dalam format MapInfo dimana terdapat 180 simpul yang digunakan untuk memilih simpul-simpul penawaran dan permintaan. Pada tahap evaluasi, masukan simpul-simpul penawaran dan permintaan serta jumlah persediaan komoditas untuk setiap penawaran dan dan jumlah kebutuhan untuk setiap permintaan dilakukan secara acak.

Masukan biaya diperoleh dari jarak terpendeknyang ditempuh dari setiap simpul penawaran ke simpul permintaan dengan menggunakan algoritma A*. Solusi awal dari permasalahan transportasi diperoleh dengan menggunakan algoritma genetika, yang diawali dari pembentukan populasi awal yang kemudian dilakukan pemilihan dua *parent* secara acak melalui prosedur *roulette wheel* yang digunakan untuk pembentukan *offspring* melalui operasi *crossover* dan *mutation*. Solusi awal masalah transportasi linier diperoleh dari pemilihan nilai *fitness* terkecil dari *parent* yang ada.

Evaluasi algoritma genetika

1. Pemilihan parameter : kombinasi dari jumlah iterasi, populasi awal, probabilitas operasi *crossove*r dan *mutation*

   Kombinasi : 16

   Percobaan : 3 masalah transportasi mxn (5x5, 15x20 dan 25 x30) dalam 5 kali percobaan

   Hasilnya :

   a. solusi awal masalah transportasi semakin kecil dengan menambah jumlah iterasi, jumlah populasi awal dan probabilitas operasi crossover
   b. untuk menemukan solusi awal masalah transportasi dengan hasil minimal dibutuhkan waktu yang lebih besar

2. Evaluasi hasil solusi awal dibandingkan dengan VAM

   Hasilnya :

   a. Hampir sama, apabila jumlah iterasinya tidak banyak
   b. Lebih baik, apabila jumlah iterasinya lebih banyak
   c. Lebih buruk, apabila jumlah waktunya sama
   d. Algoritma genetika menemukan solusi awal yang layak

3. Evaluasi waktu pencarian solusi awal terhadap jumlah kromosom

   Hasilnya : waktu yang dibutuhkan untuk menemukan solusi awal menggunakan algoritma genetika akan mengikuti kurva persamaan kuadrat

4. Evaluasi jumlah iterasi dan waktu

   Hasilnya : .untuk kenaikan ukuran percobaan sebanyak 50, jumlah percobaan yang dibutuhkan mengalami kenaikan sebesar 10 juta percobaan dan kenaikan waktu untuk 10 kalinya. Kenaikan waktu sangat tergantung pada kecepatan komputer sehingga kenaikan waktu. Dari solusi awal yang dihasilkan menunjukkan hasil algoritma genetika lebih baik dari Vogel.

KESIMPULAN

Dari penelitian yang dilakukan, maka dapat diambil kesimpulan bahwa

1. Parameter yang tepat untuk penerapan algoritma genetika pada masalah transportasi adalah jumlah iterasi, jumlah populasi awal dan probabilitas operasi *cross over* yang besar, sedangkan probabilitas operasi *mutation* kecil.
2. Dengan algoritma genetika selalu diperoleh solusi awal yang layak dan waktu yang diperlukan untuk menemukan solusi awal akan mengikuti kurva persamaan kuadrat. Selain itu, solusi awal yang dihasilkan dengan algortima genetika lkan ebih baik dari pada Vogel apabila percobaan dilakukan dengan jumlah iterasi yang lebih banyak.

DAFTAR PUSTAKA

APN, N., Suyanto, Thaib, D. 2002. *Implementasi Algoritma Genetika Klasik Untuk Menyelesaikan Masalah Transportasi Linier Pada Sistem Manufaktur Dengan Menggunakan Algoritma A* Sebagai Masukan Jarak/Cost'*Tugas Akhir. Bandung:STT-Telkom

Bazaraa. M., Jarvis, J. and Sherali, H. 1990. *Linear Programming and Networks Flows 2nd ed*. New York : John Wiley and Sons

Gen, M & R. Cheng.1997. *Genetic Algorithms & Engineering Design*. New York: John Wiley and sons.

Siang, J.J.2011..*Riset Operasi Dalam Pendekatan Algoritmis*. Yogyakarta: Andi.

Taha, H.A. 1992. Operation Research 5th ed. New York : Mac Millan Publishing Co.

……,(http://adit279.wordpress.com/2008/12/04/*implementasi-algoritma-a-pada-permasalahan-optimasi-solusi-dynamic-water-jug/*)

Memberikan

SERTIFIKAT

kepada

Dina Thaib

sebagai

Pemakalah

pada Seminar Nasional Tahunan Matematika, Sains, dan Teknologi FMIPA-UT 2012 dengan tema "Meningkatkan Kemandirian Masyarakat dalam Pengelolaan Energi secara Bijak melalui Penerapan Matematika, Sains, dan Teknologi yang Inovatif" yang diselenggarakan pada 10 September 2012 di Balai Sidang Universitas Terbuka (UTCC)

Tangerang Selatan, 10 September 2012
Dekan FMIPA-UT

Dr. Nuraini Soleiman, M. Ed
NIP. 19540730 198601 2 001