

# **PENENTUAN RUTE PENDISTRIBUSIAN SURAT KABAR DENGAN *TIME WINDOW*, APLIKASI ALGORITMA TABU SEARCH (STUDI KASUS : KORAN KOMPAS)**

Herodia Adi Kuncoro, Ira Prasetyaningrum S.Si,MT., Renga Asmara S.KOM, OCA.

*Jurusan Teknik Informatika, PENS - ITS Surabaya*

*Jl. Raya ITS, Surabaya*

*+62(31) 594 7280; Fax: +62(31) 594 6114*

**E-mail :** [herodia@student.eepis-its.edu](mailto:herodia@student.eepis-its.edu), [ira@eepis-its.edu](mailto:ira@eepis-its.edu), [rengga@eepis-its.edu](mailto:rengga@eepis-its.edu)

## **ABSTRAK**

*Di dalam tugas akhir ini akan dibuat sebuah aplikasi untuk menentukan rute yang paling baik dan efisien dengan parameter jarak, waktu dan kuota dimana nantinya aplikasi ini akan digunakan untuk mempermudah seorang distributor surat kabar untuk menentukan rute pengiriman ke agen – agen yang ada di seluruh surabaya dimana di setiap agen telah ditentukan kapan batas maksimal seorang distributor untuk sampai di sebuah agen. Sehingga dengan aplikasi ini akan didapatkan banyak kendaraan distributor yang akan berangkat ke agen dan tidak ada distributor yang datang terlambat ke tempat agen, dengan aplikasi ini pada akhirnya akan ditampilkan visualisasi menggunakan google map dari rute terbaik yang berhasil didapatkan oleh perhitungan tabu search dengan menggunakan parameter waktu dan kuota pada kendaraan.*

**Kata Kunci** *Tabu Search, Time Window, Penentuan Rute*

## **1. PENDAHULUAN**

Koran adalah suatu surat kabar berita yang banyak digunakan sebagai pemuat semua berita tentang segala bidang. Koran selalu terbit setiap hari dan banyak orang berlangganan koran untuk selalu tahu tentang informasi terkini yang tengah terjadi. Banyaknya pelanggan koran di suatu kota merupakan masalah tersendiri yang harus bisa diatasi oleh distributor koran tersebut, sehingga permintaan para konsumen yang berlangganan koran dapat terpenuhi. Begitu juga pendistribusian ke agen – agen pengecer yang juga tidak kalah pentingnya karena menjual langsung ke konsumen, dan pendistribusian ke agen haruslah dilakukan pagi – pagi benar karena pada pagi harilah banyak konsumen yang mencari koran untuk mengetahui kabar yang terjadi pada hari sebelumnya.

Melihat dari keadaan ini maka parameter waktu menjadi sangat penting dalam hal ini, sehingga distributor harus mengerti jalur pengiriman yang paling cepat ke agen penjual beserta pertimbangan waktu yang akan dihabiskan di jalan sehingga dapat diperkirakan kapan seorang distributor berangkat dari pusat menuju ke agen – agen tersebut dan tentu saja sampai tujuan tepat sebelum pasar sedang ramai,

karena apabila sampai telat dari waktu tersebut maka dapat dipastikan penjualan akan menurun.

Untuk mengatasi hal itu maka akan dibuat suatu aplikasi untuk melakukan penghitungan jarak yang ada dari setiap letak agen yang ada di kota surabaya sehingga nantinya dapat diharapkan membantu para distributor untuk melakukan pengiriman barang ke agen – agen terdekat dengan melalui rute tercepat hasil dari penghitungan dari aplikasi ini dengan metode tabu search, dan disini juga memakai konsep VRPTW (Vehicle Routing Problem with Time Windows) yaitu sebuah konsep untuk memperkirakan waktu tempuh yang dibutuhkan seorang distributor, sehingga ada 3 parameter yang sangat penting yaitu kuota, waktu dan jarak yang harus diperhatikan oleh seorang distributor dalam pengiriman barang, dimana seorang distributor tidak boleh terlambat untuk datang ke tempat agen begitu juga harus memperhatikan kuota yang dibutuhkan oleh setiap agen dan tentu saja tanpa memberikan beban yang berlebihan pada kendaraan karena tiap kendaraan memiliki kuota maksimal yang tidak boleh di lebihkan karena akan mengganggu proses pengiriman. Dalam proses VRP pun memiliki sebuah aturan khusus dimana setiap distributor hanya boleh

datang ke tempat agen untuk melayani permintaan hanya satu kali karena ada batasan maksimal kuota pada sebuah kendaraan.

#### **Batasan masalah pada proyek akhir ini adalah :**

Batasan masalah dalam pembuatan aplikasi distribusi ini antara lain:

1. Daerah yang digunakan sebagai objek adalah kota Surabaya.
2. Jalan yang akan dilewati adalah rekomendasi dari googlemaps
3. Data Agen yang digunakan terupdate terakhir 10 agustus 2010.
4. Data diperoleh dari PT Kompas, dimana data tersebut meliputi data :
  - a. Letak agen yang tersebar di kota Surabaya.
  - b. Waktu tempuh statis tanpa ada hambatan (tetap).
  - c. Kuota setiap agen yang ada di kota Surabaya.
  - d. Setiap kendaraan dalam pengiriman surat kabar ke agen memiliki kuota yang sama.
5. Distributor harus sampai pada agen pada waktu yang ditentukan.

## **2. PENELITIAN YANG TERKAIT**

Komang Setemen, dalam makalahnya yang berjudul “Kombinasi Algoritma Genetika dan Tabu Search dalam Pembuatan Tabel Jadwal Mata Kuliah” menjelaskan bagaimana proses pengolahan penjadwalan di Surabaya dan apa saja infrastrukturnya. Dia menyebutkan bahwa Sistem umum manajemen penjadwalan Surabaya saat ini melingkup area jadwal kuliah dan penempatan kelas. Pada kasus atau penelitian ini, *tabu search* hanya digunakan untuk memfilter kromosom yang mengalami *crossover* agar kromosom yang sama tidak dilakukan *crossover* berulang-ulang. Saat iterasi pertama kali, semua kromosom yang mengalami *crossover* disimpan ke dalam *tabu list*, kemudian baru dilakukan *crossover*. Untuk iterasi berikutnya, kromosom yang mengalami *crossover*, terlebih dahulu di cek pada *tabu list*, apakah kromosom tersebut sudah ada atau belum. Jika kromosom tersebut sudah ada pada *tabu list*, maka kromosom tersebut akan dilakukan mutasi, jika belum maka kromosom tersebut disimpan pada *tabu list*, kemudian dilakukan *crossover*.

Betrianis dan Putu Teguh Aryawan dengan makalahnya ‘ Penerapan Algoritma Tabu Search Dalam Penjadwalan Job SHOP’ *Tabu Search*

merupakan salah satu metode pemecahan permasalahan optimasi kombinatorial yang tergabung ke dalam *local search methods*. Metode ini bertujuan untuk mengefektifkan proses pencarian solusi terbaik dari suatu permasalahan optimasi kombinatorial yang berskala besar (bersifat *np-hard*), contohnya permasalahan penjadwalan *job shop*, dengan waktu komputasi yang relatif lebih kecil, namun tanpa ada jaminan akan tercapainya solusi yang optimal. Dalam penelitian ini, *Tabu search* diterapkan pada sebuah permasalahan penjadwalan *job shop* dengan tujuan untuk meminimalkan waktu proses total atau *makespan* ( $C_{max}$ ). Penjadwalan menggunakan algoritma *Tabu Search* ini dilakukan terhadap tiga kasus, yaitu paket pesanan bulan September, Oktober dan Nopember, dimana untuk setiap paket pesanan dilakukan variasi terhadap *initial solution* dan panjang *tabu list*. Hasil penjadwalan ini kemudian dibandingkan dengan hasil penjadwalan lain yang menggunakan 4 macam metode *basic dispatching rules*, yaitu *Shortest Processing Time (SPT)*, *Earliest Due Date (EDD)*, *Most Work Remaining (MWKR)* dan *First Come First Served (FCFS)*. Hasil pengolahan data menunjukkan bahwa penjadwalan yang menggunakan algoritma *Tabu Search* sensitif terhadap perubahan yang diberikan pada variabel yang ada didalamnya dan *makespan* yang dihasilkan secara keseluruhan lebih kecil apabila dibandingkan dengan hasil penjadwalan menggunakan ke-4 metode lainnya.

## **3. ALGORITMA TABU SEARCH**

Kata tabu (atau *taboo*) berasal dari Tongan, sebuah bahasa Polynesia, yang mana telah digunakan oleh orang aborigin pulau Tongan untuk mengindikasikan barang yang tidak bisa disentuh karena disakralkan. Tabu Search diperkenalkan oleh Fred Glover pada tahun 1986, yang setuju dengan metode Local Search untuk memecahkan masalah local optima. Prinsip dasar tabu search adalah untuk mengikuti kemampuan local search bertemu sebuah local optimum dengan cara membiarkan nonimproving bergerak kembali ke solusi sebelumnya yang dicegah dengan menggunakan memori yang disebut dengan Tabu List, yang merekam sejarah terbaru, sebuah ide kunci yang bisa dihubungkan dengan konsep intelegensia buatan. Tabu List yang ada pada tabu search digunakan untuk menyimpan sekumpulan solusi yang baru saja dievaluasi. Selama proses optimasi, pada setiap iterasi, solusi yang akan dievaluasi akan dicocokkan terlebih dahulu dengan isi tabu list. Apabila solusi tersebut sudah ada pada tabu list, maka solusi tersebut tidak akan dievaluasi lagi pada iterasi berikutnya. Apabila sudah tidak ada lagi solusi yang tidak menjadi anggota tabu list, maka nilai terbaik yang baru saja diperoleh merupakan solusi yang sebenarnya.

Tabu Search mulai dengan cara yang sama seperti neighbourhood search (lokal) biasa dengan cara kerja secara iterasi dari suatu titik (solusi) ke solusi lain hingga kriteria terminasi yang telah dipilih dipenuhi.

Perekaman solusi secara lengkap dalam sebuah *forbidden list* dan pengecekan apakah sebuah kandidat solusi tercatat dalam list tersebut merupakan cara yang mahal, baik dari sisi kebutuhan memori maupun kebutuhan waktu komputasi. Jadi *tabu list* hanya menyimpan langkah transisi (*move*) yang merupakan lawan atau kebalikan dari langkah yang telah digunakan dalam iterasi sebelumnya untuk bergerak dari satu solusi ke solusi berikutnya. Dengan kata lain *tabu list* berisi langkah-langkah yang membalikkan solusi yang baru ke solusi yang lama. Metode "*neighbourhood search*" yang terkenal yang telah digunakan untuk menemukan sebuah penaksiran untuk nilai minimum dari "*realvalued function f*" pada sebuah kumpulan *S* adalah "*metode descent*", yang dijabarkan sebagai berikut:

Langkah 1: Pilih "solution *i*" awal pada *S*  
 Langkah 2: Temukan "*j*" yang terbaik pada  $N(i)$  (contoh:  
 $f(j) \leq f(k)$  untuk *k* apa saja yang ada pada  $N(i)$ )  
 Langkah 3: Hentikan jika  $f(j) \geq f(i)$ .  
 Yang lain, susun  $i=j$  dan lanjutkan ke langkah 2.

Pada tiap iterasi, dipilih solusi baru yang merupakan solusi terbaik dalam *neighborhood* dan tidak tergolong sebagai tabu. Kualitas solusi baru ini tidak harus lebih baik dari kualitas solusi sekarang. Apabila solusi baru ini memiliki nilai fungsi objektif lebih baik dibandingkan solusi terbaik yang telah dicapai sebelumnya, maka solusi baru ini dicatat sebagai solusi terbaik yang baru. Sebagai tambahan dari *tabu-list*, dikenal adanya *kriteria aspirasi*, yaitu suatu penanganan khusus terhadap *move* yang dinilai dapat menghasilkan solusi yang baik namun *move* tersebut berstatus tabu. Dalam hal ini, jika *move* tersebut memenuhi kriteria aspirasi yang telah ditetapkan sebelumnya, maka *move* tersebut dapat digunakan untuk membentuk solusi berikutnya (status tabunya dibatalkan).

#### 4. TEKNOLOGI GOOGLE MAPS API

Google Maps adalah layanan gratis Google yang cukup populer. Kita dapat menambahkan fitur Google Maps dalam web Kita sendiri dengan Google Maps API. Google Maps API adalah library JavaScript. Yang dibutuhkan dalam penggunaan Google Maps API adalah pengetahuan tentang

HTML dan JavaScript, serta koneksi Internet. Dengan menggunakan Google Maps API Kita dapat menghemat waktu dan biaya Kita untuk membangun aplikasi peta digital yang handal, sehingga Kita dapat focus hanya pada data-data. Biarkan data peta dunia menjadi urusan Google saja.

Saat ini versi terakhir Google Map API adalah versi 3. Versi ini, katanya, akan tampil lebih cepat dari versi sebelumnya khususnya untuk browser ponsel.

Untuk mulai menggunakan Google Maps API, kita bisa melakukan beberapa langkah berikut ini:

1. Memasukkan Maps API JavaScript ke dalam HTML kita.
2. Membuat element div dengan nama `map_canvas` untuk menampilkan peta.
3. Membuat beberapa objek literal untuk menyimpan property- properti pada peta.
4. Menuliskan fungsi JavaScript untuk membuat objek peta.
5. Meng-inisiasi peta dalam tag body HTML dengan event `onload`.

Contoh penggunaan Google Maps API bisa kita lihat dalam gambar berikut ini. Disini, Google Maps API digunakan untuk menampilkan marker secara acak disebuah peta.

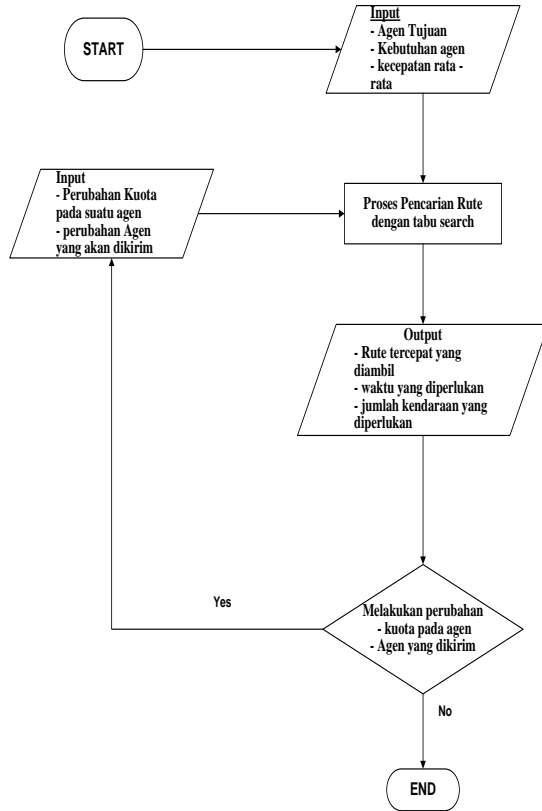


**Gambar 1:** Contoh penggunaan Google Maps API

#### 5. PERANCANGAN SISTEM

Cara kerja sistem ini akan dibagi ke dalam beberapa tahapan proses supaya dapat dilihat dengan jelas. Tahapan proses dibagi menjadi beberapa tahapan sebelum tercipta sebuah sistem yang bisa digunakan untuk mendapatkan menentukan rute pendistribusian barang.

Sistem ini terbagi menjadi dua bagian pokok, yaitu pembuatan sistem optimasi yang menggunakan Algoritma Tabu search, dengan simulasi rute yang menggunakan teknologi peta dari Google Maps API. Berikut ini adalah flowchart umum perancangan sistem optimasi yang dibuat.



**Gambar 2:** Alur umum sistem optimasi

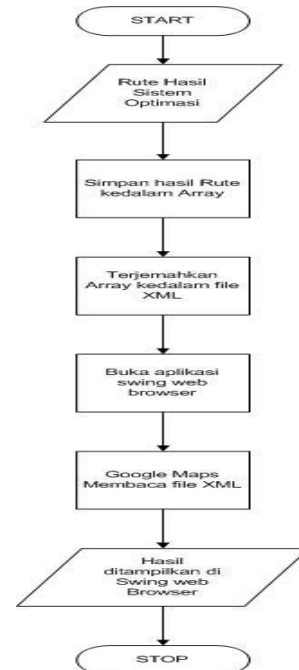
Keterangan:

1. Sistem ini mengatur jalur pengangkutan surat kabar dari Agen ke Agen -agen.
2. Untuk jalur pengangkutan, maka sistem akan memilih jalur yang baik untuk dilewati distributor ke tempat agen - agen.
3. Jalur rute yang ditempuh oleh distributor selalu berubah – ubah sesuai inputan agen mana saja yang meminta dikirim surat kabar.
4. Input dari sistem optimasi ini adalah lokasi Agen, dan demand atau kuota yang diminta oleh agen tersebut. Setelah menerima input tersebut, sistem akan memberikan output berupa jumlah kendaraan yang harus dikirimkan untuk mengirimkan surat kabar ke tempat agen – agen yang bersangkutan dan akan menampilkan rute mana saja yang harus diambil oleh seorang distributor.
5. Sistem ini dituntut untuk dinamis mengikuti jumlah kuota agen yang dapat diubah - ubah.

## 6. PERANCANGAN SIG DENGAN GOOGLE MAPS API

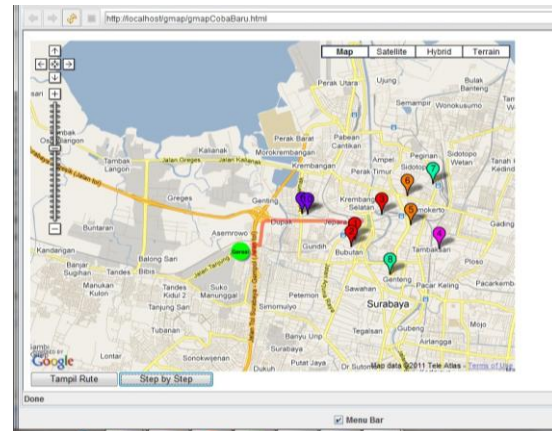
Untuk mensimulasikan rute hasil optimasi yang telah dikeluarkan oleh sistem optimasi, maka digunakan salah satu fitur dari Google Maps API

yang disebut dengan Get Direction. Karena Google Maps API menggunakan arsitektur Javascript, sedangkan program optimasi menggunakan Java Desktop, maka alur untuk mensimulasikan rute adalah sebagai berikut:



**Gambar 3:** Alur Simulasi Rute dengan Google Maps

Alur diatas dibuat karena Google Maps API tidak bisa berkomunikasi secara langsung dengan MySQL, sehingga kita harus mengkonversikannya terlebih dahulu kedalam bentuk XML. Dari XML yang kita buat itulah, kita bisa mengkomunikasikan Google Maps API dengan data hasil optimasi, sehingga Google Maps API bisa mensimulasikan rute sesuai dengan request dari sistem optimasi. Hasil dari Simulasi Sistem Optimasi contohnya adalah sebagai berikut:



**Gambar 4:** Simulasi Rute dengan Google Maps API

## 7. PERANCANGAN USER INTERFACE

### a. Desain Form Utama

NO ID Kendaraan	ID Kendaraan	Nama Terminal	Kuota Busa	Stabas Waktu
1	terminal01	gunung sari	50	04:15:00
2	terminal02	darma	60	04:20:00
3	terminal03	karang menjangan	50	04:40:00
4	terminal04	kmlaten	30	05:00:00
5	terminal05	jemursari	30	05:15:00
6	terminal06	ngagel	40	04:55:00
7	terminal07	herjages	25	05:10:00
8	terminal08	lewang	40	04:15:00
9	terminal09	darma kali	30	04:20:00
10	terminal10	dharmasuda	60	04:40:00
11	terminal11	diorejo	40	05:00:00
12	terminal12	wonosono	30	05:15:00
13	terminal13	karang pilang	40	04:55:00
14	terminal14	berowo	25	05:10:00
15	terminal15	bandas	35	04:15:00
16	terminal16	lambi kemp	30	04:25:00
17	terminal17	lakarsanti	40	04:45:00
18	terminal18	jambangan	30	05:00:00
19	terminal19	gubung	40	05:15:00
20	terminal20	lungkut	25	04:55:00

depot	Titik 1	Titik 2	Titik 3	Titik 4

Maksimal Iterasi:

Kecepatan Rata - Rata:

Kuota Truk:

Waktu Berangkat:

VRP KENDARAAN

Gambar 5: Desain Form Utama

Form ini digunakan untuk mengakomodasi semua transaksi yang ada. Di Form ini, user bisa meng-inputkan parameter-parameter yang digunakan, seperti jumlah Agen yang akan dilayani, kapasitas truk yang digunakan, waktu berangkat dari depot menuju agen – agen, hingga maksimal iterasi sebagai penghenti perhitungan.

### b. Desain Form Master

depot	id	Tempat	Kuota	Waktu
terminal01	1	gunung sari	50	04:15:00
terminal02	2	darma	60	04:20:00
terminal03	3	karang menjangan	50	04:40:00
terminal04	4	kmlaten	30	05:00:00
terminal05	5	jemursari	30	05:15:00

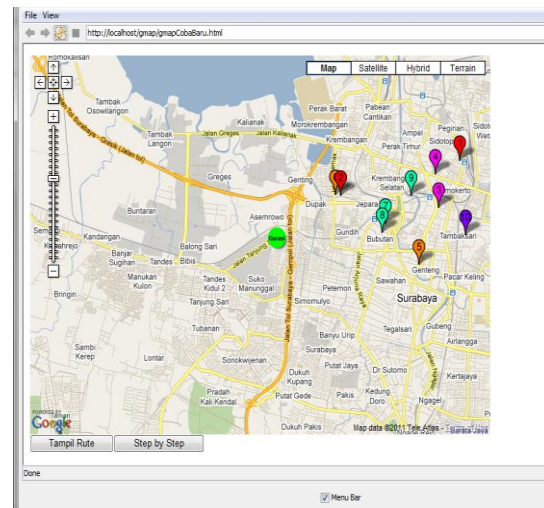
Tampil

Remove

Gambar 6: Desain Form Master

Form master ini digunakan untuk meng-inputkan berbagai macam data master, seperti kuota permintaan suatu agen, dan batas waktu seorang distributor harus sampai di agen.

### c. Desain Panel Swing Web Browser



Gambar 7: Desain Panel Swing Web Browser

Panel ini digunakan untuk menampilkan simulasi rute yang dihasilkan oleh sistem oprimasi. Panel ini menjembatani antara aplikasi simulasi rute yang dihasilkan oleh Google Maps API via Javascript dengan sistem optimasi yang berbasis Java Desktop. Panel ini menggunakan engine web browser milik Internet Explorer.

## 8. HASIL UJI COBA

Uji coba dilakukan dengan menggunakan sampel data 5 titik Agen sebagai berikut, dan dilakukan dalam 5 kali percobaan untuk diambil hasil terbaiknya:

### Data Uji Coba 5 Agen

Percobaan dilakukan dengan menggunakan beberapa parameter sebagai berikut:

### Parameter Uji Coba

Jumlah Agen = 5 , kecepatan rata – rata= 60, Kuota truk = 150 karena untuk

Ketika dilakukan ujicoba selama sepuluh kali, hasil yang didapatkan bisa dilihat dalam tabel berikut:

Tabel 1: Hasil Uji Coba

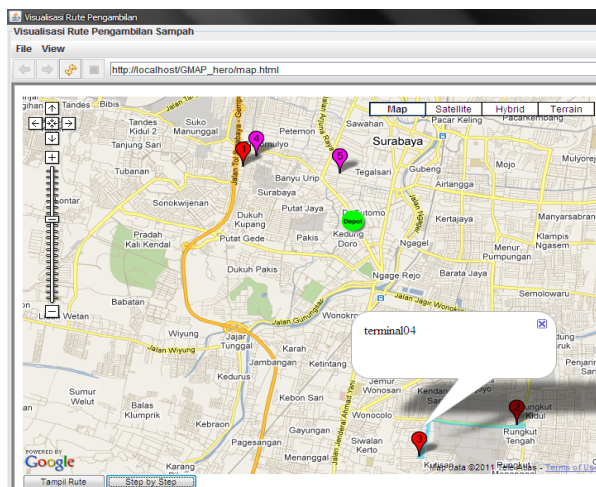
ke -	Jumlah iterasi	Banyak kendaraan	Total jarak
1	10	3	74.80
2	10	3	72.731
3	10	3	72.731
4	10	3	72.731
5	10	3	72.731

Dari percobaan diatas ,5 kali percobaan dengan menggunakan data dan parameter iterasi 10 kali mendapatkan hasil yang lebih kecil dan lebih stabil dari pada percobaan sebelum - sebelumnya dan memang pada saat percobaan 1 hasilnya memang kurang baik dibanding hasil yang setelahnya, disini kita dapat melihat bahwa memang random awal memang dapat mempengaruhi hasil dari suatu proses dalam pencarian suatu hasil dengan menggunakan tabu search solusinya adalah dengan memperbanyak iterasi yang dilakukan oleh tabu search untuk pencarian solusi akhir sehingga dapat menemukan sebuah solusi yang stabil dibandingkan dengan solusi – solusi yang menggunakan iterasi yang lebih kecil karena dari percobaan yang pernah dilakukan dengan menggunakan iterasi yang lebih kecil sebuah solusi dengan agen yang tidak sedikit akan menghasilkan sebuah hasil yang tidak stabil atau mengalami banyak perubahan dari solusi pada percobaan satu dan percobaan selanjutnya



**Gambar 8:** Rute Terbaik dari 5 agen

Setelah itu, kita tampilkan hasil terbaik tersebut kedalam Google Maps. Karena memang menampilkan rutenya secara bergantian, maka disini saya hanya meng-capture hanya satu rute saja, dari satu titik ke titik lainnya.



**Gambar 13:** Simulasi Rute dengan Google Maps

## 9. KESIMPULAN

Dari hasil uji coba sistem ini dapat ditarik beberapa kesimpulan:

1. Algoritma Tabu Search dapat digunakan untuk menyelesaikan permasalahan dalam pencarian rute terbaik untuk pengangkutan surat kabar. Sebagai pengembangan dari TSP (Travelling Salesman Problem), dan menjurus ke VRP (Vehicle Routing Problem) Algoritma Tabu Search bisa melakukan optimasi rute pengangkutan Surat kabar ke agen - agen.
2. Pembangkitan individu random pada populasi awal sangat berpengaruh pada waktu yang dibutuhkan dan jumlah iterasi yang dicapai hingga menemukan solusi.
3. Berdasarkan hasil percobaan Algoritma Tabu search cocok untuk mendapatkan solusi optimal.
4. Cara untuk mengatasi agar hasil yang dicapai tidak berubah – ubah atau stabil adalah dengan menggunakan iterasi yang besar walaupun membutuhkan resource yang besar dengan menggunakan iterasi ini maka akan lebih besar kemungkinan untuk penukaran noe secara menyeluruh sehingga walaupun random awal pertama sangat jauh dari global optimum namun kemungkinan besar setelah penukaran node – node yang menyeluruh akan menemukan solusi global optimum.

## 10. DAFTAR PUSTAKA

- [1] Kusumadewi, S. & Purnomo, H. *Penyelesaian Masalah Optimasi dengan Teknik-teknik Heuristik*. Yogyakarta: Graha Ilmu. 2005.
- [2] Google Maps API Tutorial Econym. <http://econym.org.uk/gmap/index.htm>. 14 Juli 2011
- [3] Setemen Komang, *Kombinasi Algoritma Genetika dan Tabu Search dalam Pembuatan Tabel Jadwal Mata Kuliah*, ITS: 2008.
- [4] [http://en.wikipedia.org/wiki/Tabu\\_search](http://en.wikipedia.org/wiki/Tabu_search) .15 juli 2011
- [5] <http://priyandari.staff.uns.ac.id/200909/tabu-search-introduction/> .15 juli 2011