

# ***Greedy Random : Algoritma Optimalisasi Capacitated Vehicle Routing with Time Windows***

Miridhani Riani Ningrum<sup>1</sup>, Dini Rahmawati<sup>2</sup>, Febrian Aris Rosadi<sup>3</sup>

*Laboratorium Ilmu dan Rekayasa Komputasi  
Departemen Teknik Informatika, Fakultas Teknologi Industri, Institut Teknologi Bandung (ITB)  
Kampus ITB Jl Ganesha No. 10 Bandung*

[if13023@students.if.itb.ac.id](mailto:if13023@students.if.itb.ac.id)<sup>1</sup>, [if13031@students.if.itb.ac.id](mailto:if13031@students.if.itb.ac.id)<sup>2</sup>,  
[if13041@students.if.itb.ac.id](mailto:if13041@students.if.itb.ac.id)<sup>3</sup>

## **Abstrak**

Kajian ini membahas tentang sebuah teknik untuk mengoptimalkan *Capacitated Vehicle Routing with Time Windows*. Dalam pendekatan ini sepuluh algoritma, masing-masing diimplementasikan sebagai *engine Capacitated Vehicle Routing with Time Windows*. Setelah pengimplementasian ini, diambil sebuah eksperimen mendasar yang membandingkan algoritma-algoritma tersebut satu sama lain, dan pada akhirnya diperoleh kesimpulan tentang algoritma yang terbaik, *Greedy Random*, yang menampilkan kinerja yang secara signifikan lebih baik dibandingkan sembilan algoritma lainnya. Terdapat empat buah eksperimen lainnya yang menguatkan kesimpulan tentang kesuksesan *Greedy Random*. Analisis dari eksperimen-eksperimen ini memberikan kesimpulan bahwa kesuksesan *Greedy Random* ini diperoleh dari perannya sebagai suatu *single catalytic initial* gerak acak untuk memodifikasi solusi, reoptimalisasi, serta pendekatannya ke solusi akhir yang optimal. Berdasar penelitian yang menjadi referensi, *Greedy Random* ini adalah algoritma optimalisasi independent pertama yang sukses dalam memecahkan masalah.

Kata kunci : *Capacitated Vehicle Routing with Time Windows, Greedy Random*

## **Pendahuluan**

Tujuan dari perutean kendaraan adalah untuk menjadwalkan pengantaran atau pengiriman produk kepada banyak pelanggan. Kajian ini membahas tentang *Capacitated Vehicle Routing with Time windows*, yaitu perutean kendaraan di mana masing-masing kendaraan tersebut membawa suatu produk tertentu untuk diantarkan kepada banyak pelanggan yang berbeda-beda dengan berbagai *constraint(time windows)* dan produk dengan kuantitas sesuai dengan permintaan. Dengan memasukkan *constraint* dan kuantitas permintaan produk, *Capacitated Vehicle Routing with Time Windows* meng-generate solusi yang memiliki aplikasinya dalam dunia nyata [4].

*Capacitated Vehicle Routing with Time Windows* adalah salah satu contoh *NP-Hard problem*, yaitu suatu problem atau permasalahan yang jauh lebih

kompleks dari permasalahan lain yang dapat diselesaikan dengan kompleksitas polinomial[2]. Kompleksitas disini disebabkan karena jumlah solusi yang mungkin untuk permasalahan ini meningkat sangat drastis seiring dengan penambahan ukuran permasalahan.

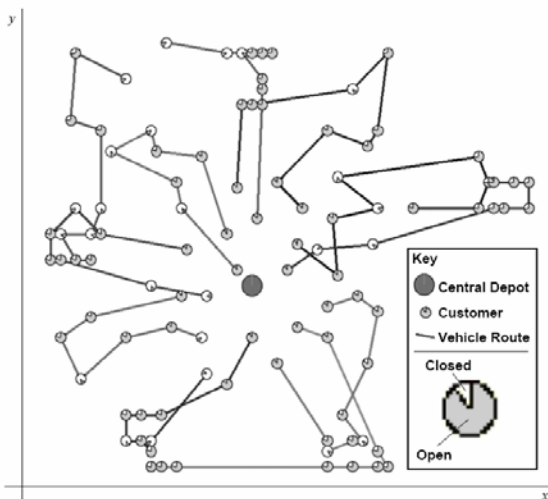
Dalam paper Solomon (1987) yaitu inialisasi penelitian terhadap *Capacitated Vehicle Routing with Time Windows* telah dikemukakan suatu standard untuk perbandingan optimalisasi[11]. Dengan menggunakan standard perbandingan tersebut, beberapa teknik telah berhasil memecahkan masalah optimalisasi *Capacitated Vehicle Routing with Time Windows* ini, termasuk *genetic algorithms*[12], *tabu search*[6], *probabilistic searches*[10], *constraint programming*[5], *exact algorithms*[3], *metaheuristic*[7], *multiple improvement heuristic*[9], dan *Human-Guided Simple Search* [1], manusia dilibatkan bersama

computer dalam pencarian solusi optimal sehingga kinerja *engine* dari *Capacitated Vehicle Routing with Time Windows* dalam pencarian solusi optimum dikontrol dan dikendalikan manusia[1].

Paper ini menguji tentang sebuah pandangann baru dalam proses optimalisasi dengan pemisahan *engine Capacitated Vehicle Routing with Time Windows* dari algoritma optimalisasinya. Pengujian pandangan baru ini dimulai dengan menggantikan posisi manusia dengan 10 macam algoritma yang berbeda. Tiap algoritma ini menggunakan teknik yang berbeda dlam mengendalikan *engine* dari *Capacitated Vehicle Routing with Time Windows*.

## Ruang Lingkup

### 1. Solusi *Capacitated Vehicle Routing with Time Windows* [1]



Gambar 1. Visualisasi dari *Capacitated Vehicle Routing with Time Windows* solusi

Lingkaran besar di tengah merepresentasikan suatu terminal di trmpat darimana kendaraan berangkat. Linkaran-linkaran lain yang lebih kecil merepresentasilam para pelanggan dan garis-garis penghubung merepresentasikan rute kendaraan. Lingkaran kecil yang menyatakan pelanggan (3) sekaligus merupakan diagram-diagram lingkaran yang menyatakan tingkat waktu (*time windows*) para pelanggan. Untuk mengurangi kerumitan atau kerugian gambar maka garis dari dan ke lingkaran terminal tidak digambarkan.

Solusi dari sebuah *Capacitated Vehicle Routing with Time Windows* adalah suatu himpunan dari  $n$  buah

pelanggan. Setiap pelanggan  $p$  mempunyai sebuah lokasi  $(X_p, Y_p)$ , tenggat waktu  $[tp_{open}, tp_{close}]$  dan jumlah permintaan produk  $J_p$ .  $V$  buah kendaraan melayani ke- $n$  buah pelanggan diatas. Setiap kendaraan  $k$  membawa sejumlah  $c$  produk dan melalui total jarak  $d_k$  (dari terminal ke seluruh lokasi pelanggan dan kembali lagi ke terminal). Setiap pelanggan mendapat produk sesuai perminataan ( $J_p$ ) dalam tenggat waktu  $[tp_{delivery}]$ .

Suatu solusi adalah jika  $V_p, tp_{delivery} \in [tp_{open}, tp_{close}]$  dan  $\forall p, \forall p \in \{p_1, p_2, \dots, p_n\}, ((\sum J_p) < c)$  (yaitu semua pelanggan menerima produk sesuai permintaan dalam tenggat waktu yang sesuai dan tidak ada kendaraan yang kehabisan produk). Jika kondisi diatas tidak terpenuhi, maka solusi tersebut tidak mungkin.

*Cost* (beban) dari suatu solusi adalah jumlah kendaraan ( $k$ ). Jika  $k$  bernilai sama untuk 2 buah solusi, maka jumlah  $\sum d_k$  (jumlah total jarak yang ditempuh kendaraan) digunakan untuk membedakannya. Suatu solusi yang tidak mungkin memiliki *cost* yang tak terhingga.

*Goal* (hasil akhir) dari optimalisasi ini adalah untuk menentukan suatu solusi yang memiliki *cost* terkecil.

### 2. *Engine Capacitated Vehicle Routing with Time Windows*

Setiap algoritma bekerja pada suatu interval waktu diskrit yang disebut *cycles*. Dalam sebuah *cycles*, algoritma membangkitkan *engine Capacitated Vehicle Routing with Time Windows* untuk mengoptimalisasi solusi. Secara sistemati, *engine Capacitated Vehicle Routing with Time Windows* akan mencari melalui semua jalan yang mungkin untuk mengurangi *cost* solusi.

*Engine Capacitated Vehicle Routing with Time Windows* mempunyai dua buah metode pencarian utama, yaitu :

1. *Greedy search*
2. *Steepest Search*

Kedua metode ini memberikan perbedaan yang jelas tentang bagaimana *engine Capacitated Vehicle Routing with Time Windows* menerapkan kemajuan (yang diperoleh dari langkah yang dijalankan) yang ditemukan.

Jika suatu *engine Capacitated Vehicle Routing with Time Windows* menggunakan metode *Greedy search*, *engine* akan langsung mengimplementasikan kemajuan yang ditemukan dan memulai kerja *engine* kembali dengan solusi yang baru. Sedangkan jika

suatu *engine Capacitated Vehicle Routing with Time Windows* menggunakan metode *Steepest Search* maka *engine* tidak akan mengimplementasi kemajuan sama sekali sampai tercapai batas waktu akhir. Setelah itu barulah *engine* akan menerapkan gerakan yang akan meminimalisasi biaya solusi.

Tetapi dari metode pencarian apa yang digunakan, *engine Capacitated Vehicle Routing with Time Windows* akan selalu berusaha untuk meminimalisasi jumlah kendaraan dan jumlah jarak yang ditempuh.

Dalam *Human-Guided Simple Search* telah didefinisikan suatu prioritas bagi pelanggan untuk memfokuskan pencarian. Prioritas ini mengacu oleh lingkup pencarian dan bukan oleh tingkat kepentingan pelanggan untuk menerima produk [1]. Ada tiga level prioritas untuk pelanggan, yaitu *high*, *medium*, dan *low*.

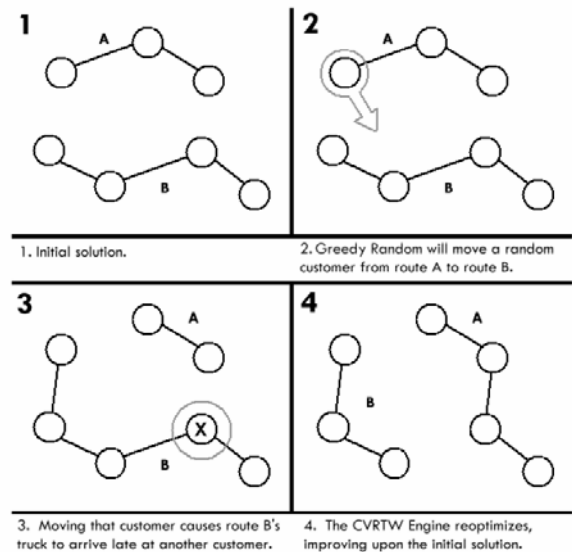
Sejalan dengan *Human-Guided Simple Search*, algoritma *Capacitated Vehicle Routing with Time Windows* juga menerapkan prioritas memfokuskan pencarian. Algoritma dapat suatu pelanggan ke prioritas *high* agar *engine Capacitated Vehicle Routing with Time Windows* dapat menggerakkan suatu pelanggan berpindah dari rute asal kendaraan ke rute lainnya. Sedangkan dengan mengeset suatu pelanggan ke prioritas *medium* atau *low* akan dapat mencegah *engine Capacitated Vehicle Routing with Time Windows* menggerakkan suatu pelanggan berpindah dari rute asalnya.

### 3. Standar perbandingan optimalisasi

Semua penelitian yang menjadi referensi berdasarkan standar *Solomon's Random Clustered (RC)*, yaitu RC101 sampai RC108.

### 4. Greedy Random

Sebelumnya, telah dilakukan suatu perjanjian terhadap 10 algoritma yang berbeda, semuanya diinformasikan dengan *engine Capacitated Vehicle Routing with Time Windows*. Hasil akhir pengujian ini menunjukkan bahwa *Greedy Random* menampilkan performansi yang lebih baik secara signifikan.



Gambar 2. Skema contoh optimalisasi *Greedy Random*

Gambar diatas menunjukkan langkah logic sekuensial dari *Greedy Random*. Pertama-tama, *Greedy Random* mengeset semua pelanggan menjadi prioritas *high*, sehingga semua kasus yang mungkin akan diperimbangkan. Lalu, secara acak (*random*) *Greedy Random* akan memilih satu pelanggan dan menggerakkannya dari suatu rute kendaraan ke rute kendaraan lainnya. Gerakan ini adalah *initial random move Greedy Random*. Kadang-kadang relokasi pelanggan ini akan membuat solusi menjadi tidak mungkin seperti pada gambar 3, tapi kadang solusi akan tetap mungkin. Dalam hal ini *Greedy Random* akan mengeset pelanggan yang bergerak tadi menjadi prioritas *medium*, sehingga *engine Capacitated Vehicle Routing with Time Windows* tidak dapat menggerakkannya lagi. Lalu *Greedy Random* akan membangkitkan *engine Capacitated Vehicle Routing with Time Windows* untuk melakukan optimalisasi pada tiap cycle (4). Jika solusi yang baru lebih baik dari solusi awal, maka solusi akhir inilah yang digunakan. Jika tidak, maka solusi akhir ini dibuang.

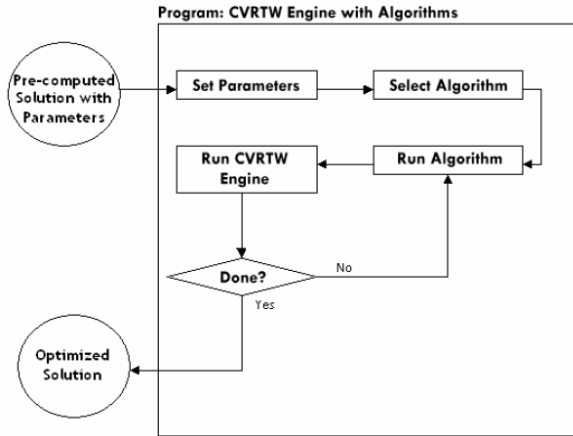
### 5. Algoritma

Selain *Greedy Random*, pengujian juga dilakukan terhadap 9 algoritma lainnya yaitu ;

- High Priority(HI)*
- Steepest Climbing (SC)*
- Random Priorities (RP)*
- Random Code Priorities (RCP)*
- Random Routes (RR)*
- Random Adjacent Routes (RAR)*
- Random Priorities Greedy Random (RPGR)*

- h. Repetitive Steepest Search (RSS)
- i. Any Algorithm (ANY)

## 6. Proses Optimalisasi



Dia *Greedy Random* alir proses optimalisasi dengan menggunakan algoritma yang mengendalikan *engine Capacitated Vehicle Routing with Time Windows*

## Pengkajian Eksperimen

### 1. Algoritma Comparison Run

Eksperimen yang pertama yaitu *Algorithm Comparison Run*. Di sini Kesepuluh algoritma yang telah disebutkan sebelumnya dibandingkan satu sama lain dengan menggunakan standar perbandingan Solomon (RC101 - RC108). Tujuan yang ingin dicapai dengan eksperimen ACR ini yaitu untuk memperoleh fakta algoritma apakah yang paling baik untuk proses optimalisasi.

Rank	Algorithm	Vehicles	Distance
1	GR	12.81	1380
2	ANY	13.13	1398
3	RSS	13.33	1393
4	SC	13.38	1398
5	RPGR	13.44	1447
6	RR	13.50	1456
7	RP	13.54	1405
8	RAR	13.65	1415
9	RCP	13.99	1457
10	HI	14.65	1572

Tabel 1. Hasil pengujian dengan Algorithm Comparison Run dan ranking masing-masing algoritma

Tabel 1 di atas mengilustrasikan hasil pengujian *Algorithm Comparison Run.. Greedy Random* menghasilkan rata-rata jumlah kendaraan yang lebih rendah dibandingkan algoritma yang lainnya. Analisis secara statistik terhadap pengujian *Algorithm Comparison Run.* menunjukkan bahwa *Greedy Random* secara signifikan lebih baik daripada algoritma lainnya.

### 2. Feasible/ Infeasible Run

*Feasible/ Infeasible Run* menghipotesiskan bahwa *Greedy Random* memperoleh kesuksesannya karena kegunaannya dalam suatu kemungkinan yang *infeasible*, suatu teknik yang tidak dapat dilakukan oleh algoritma lainnya. Dengan membandingkan dua variasi dari *Greedy Random*, *Feasible/ Infeasible Run* menunjukkan efek terhadap proses optimalisasi oleh *Greedy Random* yang diakibatkan oleh *feasibility*.

Pada variasi yang pertama, suatu *infeasible Greedy Random* hanya akan menghasilkan *initial random moves* yang *infeasible* juga. Sedangkan variasi yang *feasible* hanya akan menghasilkan *initial random moves* yang *feasible* juga.

Benchmark	Infeasible GR	Feasible GR
RC101	15:1652	15:1681
RC102	13.5:1492	14:1500
RC103	11:1375	11:1365
RC104	10:1196	10:1200
RC105	14:1566	14:1568
RC106	12:1431	12:1437
RC107	11:1271	11:1259
RC108	11:1184	11:1182
Scores	12.1875:1396	12.25:1399

Tabel 2. Hasil rata-rata dari *Feasible/ Infeasible Run*

Pada tabel 2 di atas, notasi 15:1652 menyatakan kendaraan : jarak yang ditempuh. Analisis statistik dari hasil pada tabel 2 menunjukkan bahwa tidak terdapat perbedaan yang signifikan antara *infeasible Greedy Random* dan *feasible Greedy Random*. Dari hal ini dapat disimpulkan bahwa *feasibility* bukanlah hal yang menentukan dalam keberhasilan *Greedy Random*.

### 3. Variable Priorities Run

*Variable Priorities Run* menghipotesiskan bahwa *Greedy Random* memperoleh keberhasilannya

karena kemampuannya yang dapat mengeset pelanggan ke *medium priority*. Tiga variasi yang diujicoba yaitu *High Priority Greedy Random*, *Medium Priority Greedy Random*, dan *Low Priority Greedy Random*.

Benchmark	High Priority GR	Medium Priority GR	Low Priority GR
RC101	15:1654.64	15:1658.9	15:1677.89
RC102	13.5:1482.47	14:1498.4	13.5:1512.33
RC103	11:1332.01	11:1349.82	11:1411.98
RC104	10:1174.37	10:1200.25	10:1196.15
RC105	14:1542.38	14:1558.97	14:1592.72
RC106	12:1417.42	12:1435.99	12:1434.55
RC107	11:1261.19	11:1279.26	11:1301.31
RC108	11:1174.62	11:1184.82	11:1193.59
Scores	12.1875:1379.89	12.25:1395.8	12.1875:1415.07

Tabel 3. Hasil rata-rata pengujian *Variable Priorities Run*

Dari tabel di atas dapat dilihat bahwa *High Priority Greedy Random* menunjukkan performa yang lebih baik daripada *Medium Priority Greedy Random* dan *Low Priority Greedy Random*

#### 4. Multiple *Initial Random Moves Run*

Dengan *Multiple Initial Random Moves Run*, hipotesis yang menyatakan bahwa keberhasilan *Greedy Random* ditentukan oleh banyaknya *initial random moves* yang ada diuji.

Benchmark	1 Move	2 Moves	5 Moves	7 Moves
RC101	15:1655.93	15:1663.59	15:1718.99	15:1718.99
RC102	13.5:1502.3	14:1516.82	14:1552.24	14:1554.07
RC103	11:1364.89	11:1379.72	11:1411.98	11:1411.98
RC104	10:1200.53	10:1196.61	10:1200.53	10:1200.53
RC105	14:1563.97	14:1570.9	14:1647.17	14:1647.17
RC106	12:1437.89	12:1437.89	12:1437.89	12:1437.89
RC107	11:1274.74	11:1296.28	11:1306.35	11:1306.35
RC108	11:1165.93	11:1187.78	11:1217.63	11:1217.63
Scores	12.1875:1395.77	12.25:1406.2	12.25:1436.6	12.25:1436.83

Tabel 4. Hasil rata-rata *Multiple Initial Random Moves Run*

Dari table diatas, setelah dianalisis, dapat disimpulkan bahwa ada suatu korelasi yang menunjukkan hubungan antara jumlah *initial random moves* dengan kinerja *Greedy Random*, yaitu semakin sedikit *initial random move* maka semakin baik kinerja *Greedy Random*. Namun kesimpulan ini bukan berarti bahwa tidak diperlukan adanya *initial random moves*.

#### Kesimpulan

Melalui eksperimen yang telah dilakukan oleh para ahli, ditemukan dan dianalisis sebuah algoritma *Greedy Random* yang menjadi suatu algoritma pemecahan untuk suatu masalah perutean kendaraan, khususnya adalah *Capacitated Vehicle Routing with*

*Time windows*. Dalam eksperimen tersebut diajukan suatu bukti bahwa *Greedy Random* mendapatkan keberhasilannya melalui perannya sebagai *single catalytic initial random move* yang membuat algoritma ini mempunyai kemampuan untuk membuang solusi non-optimal minimum dan mendapatkan solusi optimal. *Greedy Random* mempunyai portabilitas yang tinggi karena bentuk algoritmanya yang terpisah dari *engine* optimalisasinya. Hal ini membuat *Greedy Random* dapat diterapkan secara mudah dalam mengoptimalkan hasil dalam berbagai bidang lain.

#### Daftar Pustaka

1. D. Anderson, E. Anderson, N. Lesh, J. Marks, B. Mirtich and D. Ratajczak, *Human-Guided Simple Search.* " In 17th Nat. Conf. on Artificial Intelligence: Juli 2000, pp.209-21, 2000.
2. M. Atallah, Ed., *Algorithms and the Theory of Computation Handbook.* CRC Press, Boca Raton, FL, pp. 19-26, 1999.
3. E. Baker, *An Exact Algorithm for the Time-Constrained Traveling Salesmen Problem.* " *Operations Research* vol. 31, no. 5, Sept-Okt., pp. 938-945, 1983.
4. J. Braklow, W. Graham, S. Hassler, K. Peck and W. Powell, *Interactive Optimization Improves Service and Performance for Yellow Freight System.* " *INTERFACES* vol. 22, no. 1, Jan-Feb., pp 147-172, 1992.
5. De Backer, V. Furnon, P. Kilby, P. Prosser and P. Shaw, *Solving Vehicle Routing Problems using Constraint Programming and Metaheuristics.* " *Journal of Heuristics Special Issue on Constraint Programming*, Juli 1997.
6. B. Garcia, J. Potvin and J. M. Rousseau, *A parallel implementation of the tabu search heuristic for vehicle routing problems with time window constraints.* " *Computers & Operations Research* vol. 21, no. 9, pp. 1025-1033, 1994.
7. J. Homberger and H. Gehring, *Two Evolutionary Metaheuristics for the Vehicle Routing Problem with Time windows.* " *INFOR* vol. 37, no. 3, Aug., pp. 297-317, 1999.
8. D. Montgomery, *Design and Analysis of Experiments.* New York, John Wiley & Sons, 1984.
9. P. Prosser and P. Shaw, *Study of Greedy Random search with Multiple Improvement Heuristics for Vehicle Routing Problems.* " *University of Strathclyde Department of Computer Science, Glasgow, Scotland. Research Report 96/201*, Dec. 1996.

10. Y. Rochat and E. Taillard, *Probabilistic Diversification and Intensification in Local Search for Vehicle Routing.* *Journal of Heuristics* vol. 1, pp. 147-167, 1995.
11. M. Solomon, *Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints.* *Operations Research* vol. 35, no. 2, March-April, pp. 254-264, 1987.
12. S. Thangiah, *Vehicle routing with time windows using genetic algorithms.* *Artificial Intelligence and Robotics Laboratory, Computer Science Department, Slippery Rock University, Slippery Rock, PA. Technical Report*, 1993.
13. *Greedy Randomized Vehicle Routing*  
<http://dominik.net/research/GreedyRandom/GreedyRandom.ppt>  
 Diakses tanggal 16 Mei 2005 pukul 11.00
15. *Investigating Human-Computer Optimization*  
<http://www.merl.com/reports/docs/TR2001-39.pdf>. Diakses tanggal 16 Mei 2005 pukul 12.00
14. *Greedy Random Abstract*  
<http://dominik.net/personal/research/GreedyRandomizedrandom/GreedyRandomizedrandomabstract>. Diakses tanggal 16 Mei 2005 pukul 12.30.
15. *Grasp : An Annotated Bibliography*  
[http://www.optimizationonline.org/DB\\_FILE/2001/01/254.pdf](http://www.optimizationonline.org/DB_FILE/2001/01/254.pdf). Diakses tanggal 16 Mei 2005 pukul 12.30.
16. *Evaluating and Improving Human-Guided Simple*  
<http://dominik.net/research/hugss/main.pdf>.  
 Diakses tanggal 16 Mei 2005 pukul 13.00.