PENJADWALAN RUANG KULIAH MENGGUNAKAN *VERTEX GRAPH COLORING* DAN *SIMULATED ANNEALING*

Titus Kristanto[1)], Tutuk Indriyani[2)], Ni'matul Khoiroh[3)]
Jurusan Teknik Informatika, Institut Teknologi Adhi Tama Surabaya
Email: [1)]tintus.chris@gmail.com, [2)]tutuk223@gmail.com, [3)]imahonikma@gmail.com

ABSTRACT

In university, college scheduling is very important in the process of lectures, because the activity of lecturer and students are depending on class schedules. To solve the problem, using Vertex Graph Coloring and Simulated Annealing. At Vertex Graph Coloring, seeking vertex neighbors and no neighbors. While Simulated Annealing, looking for a room and swapped positions randomly. Merging Vertex Graph Coloring and Simulated Annealing aims to make college schedules optimally with a view hard constraints and soft constraint. Tests performed at the Department of Informatics, Institute of Technology Adhi Tama Surabaya, with create schedule from manual to computerized, which is expected can create an optimal schedule and were able to avoid hard constraint and soft constraint.

Keywords: *College Scheduling, Vertex Graph Coloring, Simulated Annealing, Hard Constraint, Soft Constraint*

ABSTRAK

Pada perguruan tinggi, penjadwalan kuliah sangat penting dalam proses perkuliahan, karena aktivitas dosen dan mahasiswa tergantung pada jadwal kuliah. Untuk mengatasi masalah, menggunakan Vertex Graph Coloring dan Simulated Annealing. Pada Vertex Graph Coloring, mencari vertex bertetangga dan tidak bertetangga. Sedangkan pada Simulated Annealing, mencari ruang dan bertukar posisi secara acak. Penggabungan Vertex Graph Coloring dan Simulated Annealing bertujuan untuk membuat jadwal kuliah secara optimal dengan melihat hard constraint dan soft constraint. Pengujian dilakukan di Jurusan Teknik Informatika, Institut Teknologi Adhi Tama Surabaya, dengan membuat jadwal dari manual menjadi komputerisasi, sehingga diharapkan dapat membuat jadwal secara optimal dan mampu menghindari hard constaint dan soft constraint.

Kata Kunci: Penjadwalan Kuliah, Vertex Graph Coloring, Simulated Annealing, Hard Constraint, Soft Constraint

PENDAHULUAN

Penjadwalan kuliah merupakan hal terpenting dalam kegiatan pembelajaran. Memikiki jadwal kuliah yang baik, berarti memiliki distribusi mata kuliah yang merata setiap hari tanpa mengalami kendala dalam proses perkuliahan. Dalam membuat jadwal kuliah tidak mudah. Beberapa aspek yang mempengaruhi yaitu penyusunan jadwal kuliah, mata kuliah, dan slot waktu. Setiap aspek memiliki kelemahan dalam penyusunan jadwal perkuliahan. Untuk mengatasi masalah tersebut, menggunakan sistem terkomputerisasi untuk simulasi jadwal kuliah. Sistem menerima semua permasalahan untuk menghasilkan solusi terbaik dalam menetapkan jadwal kuliah.

Dalam penjadwalan kuliah, membutuhkan model untuk memecahkan masalah.Menentukan model yang tepat, dapat dilakukan dalam bentuk identifikasi masalah, analisis ruang lingkup masalah, dan identifikasi variabel yang terlibat dalam pengambilan keputusan. Masalah yang dihadapi dalam penjadwalan kuliah adalah masalah kompleks yang harus dihadapi secara teratur. Aspek penyusunan jadwal kuliah, terhubung dengan aspek yang lain. Penjadwalan perkuliahan dilakukan setiap semester dengan lingkup berbeda dari masalah dalam setiap semester. Penyusunan slot waktu dilalui berdasarkan penjadwalan mata kuliah, sehingga mendapatkan jalur terpendek yang optimal. Solusi yang dibutuhkan adalah penggabungan *Vertex Graph Coloring* dan *Simulated Annealing.* Tujuannya adalah untuk mendapatkan penjadwalan yang terstruktur dan optimal.

TINJAUAN PUSTAKA

Penjadwalan

Penjadwalan merupakan proses pengembangan jadwal atau urutan proses yang diperlukan dalam persoalan [1]. Jadwal harus memenuhi beberapa persyaratan dan memenuhi keinginan semua orang yang terlibat. Penjadwalan perkuliahan adalah proses belajar mengajar dan rencana pembelajaran yang mencakup mata kuliah, dosen, slot waktu, dan ruang kelas. Penjadwalan dibuat dalam tabel sesuai dengan subyek pengajaran.

Karakteristik penjadwalan kuliah, antara lain [2] :

a. Setiap mahasiswa dapat memiliki jumlah mata kuliah yang berbeda.
b. Ketersediaan ruang kuliah sangat penting.
c. Jika ada dua ruang kuliah memiliki mahasiswa yang sama, maka ruangan tersebut tidak dapat dijadwalkan pada waktu yang sama.
d. Penjadwalan ujian.

Karakteristik penjadwalan ujian, antara lain [2] :

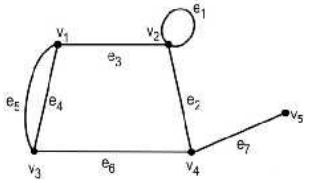
a. Hanya ada satu ujian untuk mata kuliah.
b. Adanya pembatasan, contoh pada hari sama, ada mahasiswa memiliki ujian yang banyak dan berurutan waktu.

Batasan pada penjadwalan mata kuliah, yaitu [3] :

a. *Edge Constraint*. Batasan yang mengatur dua kejadian yang tidak boleh menempati satu slot waktu yang sama.
b. *Ordering Constraint.* Batasan yang menjaga urutan kejadian secara *timeable*.
c. *Event-Spread Constraint.* Batasan yang mengatur penyebaran kejadian pada *timeable*.
d. *Present Specification and Exclusion.* Menentukan slot waktu terlebih dahulu yang digunakan sebelum proses pencarian solusi dilakukan.
e. *Capacity Constraint.* Batasan yang berhubungan dengan kapasistas ruangan.
f. *Hard Constraint.* Batasan yang tidak boleh dilanggar.
g. *Soft Constraint.* Batasan yang diusahakan semaksimal mungkin tidak boleh dilanggar, jika dilanggar masih dapat diterima.

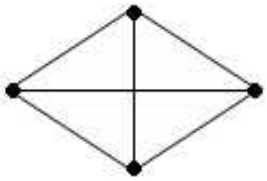
Teori Graf

Teori Graf merupakan cabang dari studi yang mempelajari sifat-sifat grafik [4]. Secara informal, graf adalah satu set obyek seperti node (simpul) yang terhubung oleh sisi (edge) atau busur. Biasanya, grafik digambarkan sebagai kumpulan titik-titik yang terhubung dengan garis atau panah. Satu sisi dapat terhubung node dengan node yang sama. Sisi tersebut dinamakan bracelet (loop). Cara mempresentasikan grafik adalah berupa diagram. Dalam diagram, ada titik-titik yang dinyatakan sebagai node dan setiap sisi dinyatakan sebagai segmen garis yang menghubungkan dua titik [5].



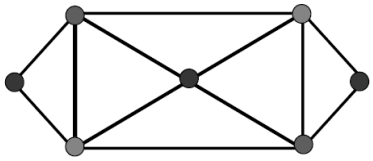
Gambar 1 Grafik dengan lima titik dan tujuh sisi

Sisi yang mempunyai dua titik akhir yang sama disebut loop. Sebuah grafik yang tidak mempunyai loop dan sisi sejajar disebut graf sederhana. Teori graf dibatasi pada graf sederhana, tetapi banyak aplikasi teknik, penggunaan loop dan sisi paralel yang diperlukan. Dalam grafik, terdapat dua sisi atau lebih yang menghubungkan dua titik yang berbeda. Kedua sisi tersebut disebut sisi ganda. Grafik yang berisi loop atau sisi rangkap disebut grafik ganda.



Gambar 2 Grafik sederhana

Definisi perwarnaan graf adalah pemberian warna, dipresentasikan sebagai bilangan urut atau diwakili langsung dengan menggunakan warna merah, biru, hijau, atau yang lain pada objek tertentu pada grafik. Objek dapat berupa simpul, sisi, wilayah, atau kombinasi ketiganya. Pada gambar 3, menjelaskan setiap node yang berdekatan atau bertetangga tidak mempunyai warna yang sama [6].



Gambar 3 Pewarnaan graf

Ada tiga macam pewarnaan graf yaitu [6] :

1) Pewarnaan Simpul *(Vertex Coloring).* Merupakan pemberian warna atau label pada setiap simpul sehingga tidak ada 2 simpul bertetangga yang memiliki warna yang sama.
2) Pewarnaan Sisi *(Edge Coloring).* Merupakan pemberian warna pada setiap sisi graf sehingga sisi-sisi yang berhubungan tidak memiliki warna yang sama.
3) Pewarnaan Wilayah *(Region Coloring).* Merupakan pemberian warna pada setiap wilayah graf sehingga tidak ada wilayah yang bersebelahan yang memiliki warna yang sama.

Vertex Graph Coloring

Graf adalah pasangan humpunan (V, E) dengan V adalah kumpulan simpul (vertex atau node), dan E adalah himpunan sisi (edge atau arc) yang menghubungkan sepasang simpul pada graf, sehingga graf diartikan sebagai kumpulan vertex yang menjelaskan hubungan antar vertex melalui edge yang terkait. Notasi vertex dan edge dituliskan dengan persamaan :

$V = \{v_1, v_2, v_3, \ldots, v_n\}$

Dan yang membentuk *graph* sebagai berikut :

$E = \{e_1, e_2, e_3, \ldots, e_n\}$

Semakin sedikit warna yang digunakan, hasil yang diperoleh semakin optimal. Masalah penjadwalan mata kuliah adalah cenderung terjadinya pelanggaran *soft constraint* yaitu pewarnaan mata kuliah lebih dari jumlah ruang kuliah yang tersedia. Sehingga mata kuliah yang tidak dijadwalkan, dapat dijadwalkan dengan melanggar *soft constraint.*

Simulated Annealing

Simulated Annealing adalah algoritma untuk optimasi yang bersifat generik [7]. Berbasiskan probabilitas dan statistik, dapat digunakan untuk mencari pendekatan terhadap solusi optimum global dari suatu permasalahan [8]. Masalah yang membutuhkan pendekatan *simulated annealing* adalah masalah optimasi kombinatorial, dimana ruang lingkup pencarian terlalu besar, sehingga tidak memungkinkan menemukan solusi yang tepat terhadap permasalahan [9]. Ada empat hal dalam penggunaan *simulated annealing* untuk memodelkan suatu permasalahan yaitu [10] :

a. Representasi akurat dari konfigurasi dalam suatu permasalahan.
b. Proses modifikasi, langkah acak atau perubahan yang harus dilakukan terhadap elemen-elemen konfigurasi untuk menghasilkan konfigurasi berikutnya.
c. Fungsi evaluasi atau fungsi objektif yang dapat menyatakan baik-buruknya solusi terhadap permasalahan.

d. Jadwal penurunan suhu dalam proses *annealing* dan berapa lama proses dilakukan.

Parameter awal yang digunakan algoritman *simulated annealing* yaitu [10]:

a. Temperatur awal, merupakan penanda awal iterasi.
b. Temperatur akhir, merupakan batas akhir penanda iterasi yang sudah dihentikan.
c. Faktor reduksi suhu, merupakan angka yang digunakan untuk menurunkan suhu secara bertahap dan terkendali.
d. Angka replikasi (nrep), merupakan angka yang menunjukkan berapa kali loop yang harus dilakukan sebelum menurunkan suhu.
e. Cari solusi awal S menggunakan parameter awal dan metode heuristik awal yang dapat ditentukan sendiri.

METODE

Metode yang digunakan dalam pembuatan jadwal kuliah yaitu gabungan *Vertex Graph Coloring* dan *Simulated Annealing.*

Deskripsi Sistem

Sistem dibuat menggunakan dua metode yaitu *Vertex Graph Coloring* dan *Simulated Annealing.* Bertujuan untuk mengetahui seberapa efisien dan optimal dalam penggabungan kedua metode. Sistem dibuat berdasarkan pertimbangan aturan *(constraints)* yang berlaku. Pengelompokkan *constraint* terdiri dari dua jenis yaitu *hard constraint* dan *soft constraint.* Pada *hard constraint* adalah aturan harus dipenuhi dalam penjadwalan perkuliahan, sedangkan *soft constraint* adalah aturan yang tidak harus dipenuhi dalam penjadwalan ruang kuliah. Penjadwalan yang dihasilkan harus optimal untuk tidak saling berbenturan dengan aturan atau batasan tertentu.

Perancangan Sistem

Dalam membuat aplikasi, terdapat tiga komponen utama yaitu input, proses dan output. Input merupakan proses pengambilan data untuk menentukan tingkat derajat dan digambarkan pada grafik. Program diproses menggunakan *Vertex Graph Coloring* dengan menggunakan algoritma *Welsh-Powell,* menghasilkan alokasi penjadwalan mata kuliah, dosen, dan ruang kuliah. Selanjutnya penjadwalan dioptimalisasi menggunakan *Simulated Annealing.*

Proses Pewarnaan Vertex Graph Coloring

Pada proses pewarnaan graf, jumlah minimum yang digunakan untuk mewarnai graf dinyatakan dalam bilangan kromatik dengan simbol $_{X}$(G) dan dilakukan pemberian warna pada *vertex* dengan mencari *vertex* tetangga dan *vertex* tidak bertetangga. Pewarnaan simpul *(vertex coloring)* ditujukan tidak ada 2 simpul bertetangga memiliki warna yang sama. Algoritma yang digunakan dalam pewarnaan *vertex* menggunakan algoritma *Welsh-Powell.* Proses pewarnaan simpul menggunakan *Vertex Graph Coloring* dengan algoritma *Welsh-Powell* dijelaskan sebagai berikut :

a. Mengumpulkan semua simpul berdasarkan derajat, dari derajat terbesar ke derajat terkecil. Derajat terbesar dari simpul ditentukan oleh banyaknya *edge* yang terhubung pada *vertex.*
b. Melakukan pewarnaan pada simpul dimulai dari derajat terbesar dengan warna pertama. Selanjutnya diberikan warna yang sama pada simpul lain yang tidak bertetangga dengan simpul pertama.
c. Melakukan pewarnaan dengan warna lain untuk simpul bertetangga dengan simpul yang sudah diberi warna. Pewarnaan berdasarkan derajat sampai semua simpul sudah diberi warna.

Proses Optimasi

Proses optimalisasi dilakukan menggunakan *Simulated Annealing,* dengan cara memeriksa pengalokasian jadwal dengan menelusuri ruang secara acak, menukar posisi dan melihat pelanggaran yang terjadi terhadap aturan yang telah digunakan.

HASIL DAN PEMBAHASAN

Berdasarkan analisis desain sistem yang telah dilakukan, dapat menerapkan penjadwalan kelas dengan menggunakan Vertex Grafik Mewarnai dan Simulated Annealing.

Analisis Sistem

Sistem dianalisis sesuai dengan aturan berlaku. Berdasarkan hasil survey, didapatkan aturan untuk pembentukan jadwal sebagai berikut :

1) *Hard Constraint*

Merupakan aturan yang tidak boleh dilanggar, jika dilanggar menimbulkan jadwal yang berbenturan. Ada beberapa jenis dari *hard constraint* yaitu :

a) *Hard constraint* terkait dengan dosen. Dosen yang sama dapat mengajar sejumlah mata kuliah, maka jadwal kelas untuk dosen yang sama sudah tentu berbeda.

b) *Hard constraint* terkait jumlah ruang yang digunakan untuk perkuliahan. Ketersediaan ruang kelas diatur oleh sistem agar tidak terjadi benturan dalam perkuliahan.

Tabel 1 Ruang Kuliah

No	Ruang	No	Ruang
1	H1-203	7	H1-304
2	H1-204	8	H1-305
3	H1-205	9	H3-201
4	H2-202	10	H3-202
5	H2-203	11	H3-203
6	H2-204	12	H3-204

c) *Hard constraint* terkait dengan mata kuliah. Mata kuliah dalam satu semester yang sama, untuk kelas yang beda dapat dibenturkan, tetapi jika sama tidak boleh dilaksanakan dalam waktu yang sama.

d) *Hard constraint* terkait dengan slot waktu.

Tabel 2 Slot Waktu Penjadwalan

Jam (Slot)	Waktu	Keterangan
1	07.30 – 10.00	3 sks
	08.20 – 10.00	2 & 1 sks
2	10.00 – 11.40	2 & 1 sks
3	12.20 – 14.00	2 & 1 sks
4	14.00 – 15.40	2 & 1 sks
	14.00 – 16.30	3 sks
6	17.00 – 19.30	3 sks
	18.00 – 19.30	2 & 1 sks
7	19.40 – 21.20	2 & 1 sks
	19.40 – 22.10	3 sks

2) *Soft Constraint*

Merupakan aturan yang boleh dilanggar, namun apabila dilanggar, maka hasil penjadwalan yang dihasilkan lebih baik. Ada beberapa jenis *soft constraint* yaitu :

a) *Soft constraint* untuk mata kuliah pilihan yang termasuk dalam satu bidang kajian yang sama, penjadwalan tidak boleh ditabrakan.

b) *Soft constraint* untuk mata kuliah pilihan sebaiknya tidak ditabrakkan jadwal dengan mata kuliah di semester 5, 6, 7, karena pada semester tersebut mahasiswa dapat mengambil mata kuliah pilihan.

Analisis Hasil

Hasil dari penjadwalan dengan sistem harus dilakukan uji hasil dengan penjadwalan secara manual sehingga dapat dibandingkan. Berdasarkan hasil pengujian didapatkan data yaitu :

a) Pengujian Penjadwalan Dosen

Tabel 3 Pengujian Penjadwalan Dosen

Semester	Jumlah Kelas				Total	Pelanggaran				Total	Prosentase
	A	B	H	I		A	B	H	I		Keberhasilan
1	5	5	5	5	20	0	0	0	0	0	100 %
2	5	5	5	5	20	0	0	0	0	0	100 %
3	8	8	8	8	32	0	0	0	0	0	100 %
4	8	8	8	8	32	0	0	0	0	0	100 %
5	9	9	9	9	36	0	0	0	0	0	100 %
6	13	13	15	15	56	0	0	0	0	0	100 %
7	10	10	10	7	37	0	0	0	0	0	100 %

b) Pengujian Penjadwalan Ruang Kelas

Tabel 4 Pengujian Penjadwalan Ruang Kelas

Semester	Jumlah Kelas				Total	Pelanggaran				Total	Prosentase
	A	B	H	I		A	B	H	I		Keberhasilan
1	5	5	5	5	20	0	1	0	2	3	85%
2	5	5	5	5	20	0	0	1	0	1	95%
3	8	8	8	8	32	0	1	0	0	1	96.875%
4	8	8	8	8	32	0	0	0	1	1	96.875%
5	9	9	9	9	36	0	1	1	0	2	94.4%
6	13	13	15	15	56	0	0	1	2	3	94.6%
7	10	10	10	7	37	0	1	0	1	2	94.5%

c) Pengujian Penjadwalan Mata Kuliah

Tabel 5 Pengujian Penjadwalan Mata Kuliah

Semester	Jumlah Kelas				Total	Pelanggaran				Total	Prosentase
	A	B	H	I		A	B	H	I		Keberhasilan
1	5	5	5	5	20	0	0	0	0	0	100 %
2	5	5	5	5	20	0	0	0	0	0	100 %
3	8	8	8	8	32	0	0	0	0	0	100 %
4	8	8	8	8	32	0	0	0	0	0	100 %
5	9	9	9	9	36	0	0	0	0	0	100 %
6	13	13	15	15	56	0	0	0	0	0	100 %

d) Pengujian Penjadwalan Slot Waktu

Tabel 6 Pengujian Penjadwalan Slot Waktu

Semester	Jumlah Kelas				Total	Pelanggaran				Total	Prosentase
	A	B	H	I		A	B	H	I		Keberhasilan
1	5	5	5	5	20	0	0	0	0	0	100 %
2	5	5	5	5	20	0	0	0	0	0	100 %
3	8	8	8	8	32	0	0	0	0	0	100 %
4	8	8	8	8	32	0	0	0	0	0	100 %
5	9	9	9	9	36	0	0	0	0	0	100 %
6	13	13	15	15	56	0	0	0	0	0	100 %
7	10	10	10	7	37	0	0	0	0	0	100 %

e) Pengujian terhadap *soft constraint* pada mata kuliah pilihan dalam satu bidang keahlian yang sama, penjadwalan tidak boleh dibenturkan. Pada pengujian untuk mata kuliah pilihan pada semester 6 dan semester 7.

Tabel 7 Pengujian *Soft Constraint* Berdasarkan Mata Kuliah Pilihan Semester 6 dan 7

Semester	Jumlah Kelas				Total	Pelanggaran				Total	Prosentase
	A	B	H	I		A	B	H	I		Keberhasilan
6	13	13	15	15	56	2	2	2	2	8	85,7
7	10	10	10	7	37	1	1	2	2	6	83,785

f) Pengujian terhadap *soft constraint* pada mata kuliah pilihan, sebaiknya tidak dibenturkan dengan jadwal mata kuliah semester 5, 6, 7, dikarenakan pada semester tersebut, ada mahasiswa yang mengambil mata kuliah pilihan.

Tabel 8 Pengujian *Soft Constraint* Berdasarkan Mata Kuliah Pilihan Semester 5, 6, dan 7

Semester	Jumlah Kelas				Total	Pelanggaran				Total	Prosentase
	A	B	H	I		A	B	H	I		Keberhasilan
5	9	9	9	9	36	2	3	4	4	13	63,89%
6	13	13	15	15	56	0	1	2	1	4	92,8%
7	10	10	10	7	37	2	2	1	1	6	83,78%

Berdasarkan hasil pengujian didapatkan data pelanggaran terhadap *hard constraint* dan *soft constraint* yaitu :

a) Pelanggaran pengujian terhadap *hard constraint*

Tabel 9 Pelanggaran Pengujian *Hard Constraint*

Hard Constraint	Jumlah Pelanggaran	Prosentase Keberhasilan
Ke - 1	0	100 %
Ke - 2	13	93,89 %
Ke - 3	0	100 %
Ke - 4	0	100 %
Total	13	

Berdasarkan hasil pengujian sistem didapatkan pelanggaran *hard constraint* ke-2 yaitu ruang kelas dengan total pelanggaran 13 kelas dari total jumlah kelas mulai semester 1 sampai dengan semester 7 yaitu 269 kelas, dengan rata-rata prosentase keberhasilan :

$$PK = \frac{a}{b} = \frac{100\% + 93{,}89\% + 100\% + 100\%}{4} = 98{,}47\%$$

Keterangan:
a = Jumlah Prosentasi keberhasilan
b = Jumlah *constraint*

b) Pelanggaran pengujian terhadap *soft constraint*

Tabel 10 Pelanggaran Pengujian *Soft Constraint*

Soft Constraint	Jumlah Pelanggaran	Prosentase Keberhasilan
Ke - 1	14	84,7425%
Ke – 2	23	80,16%
Total Pelanggaran	37	

Berdasarkan hasil pengujian sistem didapatkan pelanggaran *soft constraint* pertama yaitu 14 kelas dari 92 kelas dan *soft constraint* kedua sebanyak 23 kelas dari 129 kelas, dengan rata-rata prosentase keberhasilan :

$$PK = \frac{a}{b} = \frac{84{,}7425\% + 80{,}16\%}{2} = \frac{164{,}9025\%}{2} = 82{,}40\%$$

Keterangan:
a = Jumlah Prosentasi keberhasilan
b = Jumlah *constraint*

KESIMPULAN

Berdasarkan penelitian yang dilakukan, maka dapat disimpulkan sebagai berikut :

1) Menggunakan model solusi dengan *Vertex Graph Coloring* dan *Simulated Annealing* dalam pembuatan jadwal dapat menghindari kemungkinan jadwal ruang kuliah benturan, dikarenakan metode tersebut memeriksa setiap kemungkinan penjadwalan yang benturan dari dosen, ruang kelas, mata kuliah dan slot waktu yang sama. Berdasarkan hasil pengujian, rata-rata hasil pengujian berdasarkan mata kuliah 100%, rata-rata hasil pengujian berdasarkan dosen 100%, rata-rata hasil pengujian berdasarkan ruang kuliah 93,89%, rata-rata hasil pengujian berdasarkan slot waktu 100%.
2) Penjadwalan ruang kuliah dengan *vertex Graph Coloring*, mampu memenuhi aturan yang berlaku, terdiri dari *hard constraint* dan *soft constraint.* Hasil prosentase keberhasilan *hard constraint* 98,47%, pelanggaran *hard constraint* 1,53%, hasil prosentase keberhasilan *soft constraint* 82,4%, pelanggaran *soft constraint* 17,6%, sehingga dapat disimpulkan keberhasilan sistem yaitu 90,435%.
3) Sistem dapat membantu dalam penyusunan jadwal ruang kuliah terkomputerisasi, sehingga mempermudah dan mempercepat proses penyusunan penjadwalan.

REFERENSI

[1] Dian Ariani. 2011. *Optimasi Penjadwalan Mata Kuliah di Jurusan Teknik Informatika PENS dengan Menggunakan Algoritma Particle Swarm Optimization.* Institut Teknologi Sepuluh Nopember.
[2] Yelly Arviani. 2013. *Algoritma Ant Colony System Dalam Penjadwalan Kegiatan Belajar Mengajar di Sekolah Dasar.* Jurnal Ilmu Komputer. Universitas Sumatera Utara.
[3] Fang, H. 1994. *Genetic Algorithms in Timetabling and Schedulling.* Thesis. Edinburg, Scotland: University of Edinburgh.
[4] J.A. Bondy, and Murty U.S.R. 1976. *Graph Theory with Applications*. North-Holland New York Amsterdam Oxford, Elsevier Science Publishing.
[5] Gary Chartand, and Oellermann Ortrudr. 1993. *Applied and Algorithmic Graph Theory*. McGraw-Hiill, Inc.
[6] Marek Kubale. 2004. *Graph Coloring*. AMS Bookstore.
[7] Dimitris Bertsimas, and John Tsitsiklis. 1993. *Simulated Annealing*. Statictica Science.
[8] Benjamin W. Wan, Yixin Chen, and Tao Wang. 2007. *Simulated Annealing with Asymptotic Convergence for Nonlinear Constrained Optimization*. Journal of Global Optimization Vol. 39, pp 1-37.
[9] Wiwin Suwarningsih. 2014. *Simulation of Object Movement in the Graph for Distribution Optimization Products Inter-City*. Journal Scientific of Information Technology Applied, vol. 1 no. 1, pp 6-10.
[10] S. Kirkpatrick, C.D. Gelatt, and M.P. Vecchi. 1983. *Optimization by Simulated Annealing*. Science New Series, vol. 220, no. 4598, pp. 671-680.