

**IMPLEMENTASI ALGORITMA KOLONI SEMUT
PADA PENJADWALAN MATA KULIAH
UNIVERSITAS BRAWIJAYA
(Studi Kasus Jurusan Matematika Fakultas MIPA)**

Unggul Izza Mubarak
Program Studi Ilmu Komputer
Universitas Brawijaya Malang

Edi Santoso S.Si., M.Kom
Program Studi Ilmu Komputer
Universitas Brawijaya Malang

Dian Eka R S.Si., M.Kom
Program Studi Ilmu Komputer
Universitas Brawijaya Malang

ABSTRAK

Masalah penjadwalan kuliah merupakan masalah yang sangat kompleks dan rumit dimana inti dari masalah ini adalah bagaimana menjadwalkan berbagai komponen yang terdiri dari mahasiswa, dosen, dan waktu dengan memperhatikan sejumlah batasan dan syarat tertentu. Tujuan dari penjadwalan mata kuliah yaitu meminimumkan jumlah pelanggaran *hardconstrain*. *Hardconstrain* tersebut meliputi dosen tidak boleh mengajar pada waktu yang bersamaan, mata kuliah dengan semester yang sama prodi yang sama, kelas yang sama tidak boleh dijadwalkan di waktu yang bersamaan, pecahan SKS mata kuliah tidak boleh dijadwalkan pada hari yang sama, ruang perkuliahan tidak boleh digunakan untuk mata kuliah yang berbeda pada waktu yang bersamaan. Algoritma koloni semut merupakan salah satu algoritma heuristik yang dapat digunakan untuk menyelesaikan permasalahan penjadwalan mata kuliah. Berdasarkan hasil penelitian ini tidak terdapat pelanggaran untuk *hardconstrain*. Nilai parameter algoritma yang terbaik adalah $\alpha = 3$ $\beta = 3$ $\rho = 0,5$ dan *pheromone* awal 0,01.

PENDAHULUAN

Masalah penjadwalan kuliah merupakan masalah yang sangat kompleks dan rumit dimana inti dari masalah ini adalah bagaimana menjadwalkan berbagai komponen yang terdiri dari mahasiswa, dosen, ruang, dan waktu dengan memperhatikan sejumlah batasan dan syarat tertentu (*constraint*). Misalnya, dalam satu waktu tidak boleh ada dosen mengajar di dua tempat, begitu pula dengan mahasiswa. Dalam satu waktu mahasiswa tidak boleh melaksanakan perkuliahan di dua tempat pula. Artinya, dalam pembuatan jadwal perkuliahan, harus diperhatikan faktor-faktor yang mempengaruhi pembentukan jadwal tersebut sehingga diperoleh solusi jadwal yang optimal. Salah satu algoritma yang digunakan adalah *ant colony optimization*(ACO) atau lebih dikenal dengan algoritma koloni semut. Algoritma ini mampu menghasilkan solusi yang optimal dan cepat untuk permasalahan yang sedemikian kompleks dan rumit seperti permasalahan penjadwalan kuliah. Artinya bahwa solusi optimal yang akan diperoleh adalah solusi yang mampu memenuhi keseluruhan *constraint* dari masalah penjadwalan tersebut.

Penjadwalan

Penjadwalan adalah penempatan sumber daya (*resource*) dalam satu waktu. Penjadwalan mata kuliah merupakan persoalan penjadwalan

yang umum dan sulit, dimana tujuannya adalah menjadwalkan pertemuan dari sumber daya yang ada, dimana sumber daya yang dimaksud adalah dosen pengasuh mata kuliah, mata kuliah, ruang kuliah, kelas mahasiswa, dan waktu perkuliahan (Pinedo, 1994)[1].

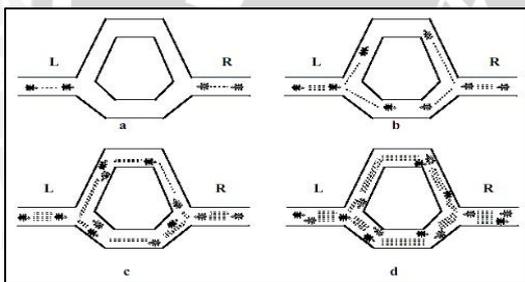
Salah satu cara untuk melakukan optimasi penjadwalan perkuliahan yang kompleks dan rumit tersebut adalah dengan menggunakan metode *heuristic*, yaitu metode yang memulai dari sebuah atau sekumpulan solusi awal, kemudian melakukan pencarian terhadap solusi yang lebih baik sehingga mendekati optimal. Kelebihan metode ini adalah tidak perlu menganalisa semua kemungkinan solusi untuk mendapatkan solusi yang optimal sehingga waktu penyelesaian yang dibutuhkan lebih cepat (Muttakhroh, 2007)[2].

Algoritma Koloni Semut

Algoritma koloni diadopsi dari perilaku semut yang dikenal sebagai sistem semut yaitu sebuah probabilistik komputasi teknik untuk memecahkan masalah yang dapat digunakan untuk menemukan jalur terbaik melalui grafik (Dorigo, 1996)[3]. Semut mampu mengindra lingkungannya yang kompleks untuk mencari makanan dan kemudian kembali ke sarangnya dengan meninggalkan zat *pheromone* pada jalur-jalur yang mereka lalui. *Pheromone*

adalah zat kimia yang berasal dari kelenjar endokrin dan digunakan oleh makhluk hidup untuk mengenali sesama jenis, individu lain, kelompok dan untuk membantu proses reproduksi. Berbeda dengan hormon, *pheromone* menyebar ke luar tubuh dapat mempengaruhi dan dikenali oleh individu lain yang sejenis (satu spesies). Proses peninggalan *pheromone* ini dikenal dengan *stigmergy*, sebuah proses memodifikasi lingkungan yang tidak hanya bertujuan untuk mengingat jalan pulang ke sarang, tetapi juga memungkinkan para semut berkomunikasi dengan koloninya (Fernandez, 2008)[4].

Gambar dibawah ini menjelaskan perjalanan semut dalam menemukan jalur terpendek dari sarang ke sumber makanan. Terdapat beberapa proses sehingga sumut sampai pada sumber makanan dan kembali lagi ke sarang.



1. Gambar diatas (a) menjelaskan ada dua kelompok semut yang akan melakukan perjalanan. Kelompok L yaitu kelompok yang berangkat dari arah kiri yang merupakan sarang semut dan kelompok R yang berangkat dari kanan yang merupakan sumber makanan.
2. Gambar diatas (b) menjelaskan kedua kelompok semut dari titik berangkat sedang dalam posisi pengambilan keputusan jalan sebelah mana yang akan diambil. Kelompok semut L membagi dua kelompok lagi. Sebagian melalui jalan atas dan sebagian melalui jalan bawah. Demikian pula dengan kelompok semut R.
3. Gambar diatas menjelaskan kelompok semut berjalan pada kecepatan yang sama dengan meninggalkan *pheromone* di jalan

yang telah dilalui. *Pheromone* yang ditinggalkan oleh kumpulan semut yang melalui jalan atas telah mengalami banyak penguapan atau evaporasi *pheromone* karena semut yang melalui jalan atas berjumlah lebih sedikit dari pada jalan yang di bawah. Hal ini dikarenakan jarak yang ditempuh lebih panjang daripada jalan bawah. Sedangkan *Pheromone* yang berada di jalan bawah, penguapannya cenderung lebih lama karena semut yang melalui jalan bawah lebih banyak daripada semut yang melalui jalan atas.

4. Gambar diatas menjelaskan semut-semut yang lain pada akhirnya memutuskan untuk melewati jalan bawah karena *pheromone* yang ditinggalkan masih banyak. Sedangkan *pheromone* pada jalan atas sudah banyak menguap sehingga semut-semut tidak memilih jalan atas tersebut. Semakin banyak semut yang melalui jalan bawah maka semakin banyak semut yang mengikutinya. Demikian juga dengan jalan atas, semakin sedikit semut yang melalui jalan atas, maka *pheromone* yang ditinggalkan semakin berkurang bahkan hilang. Dari sinilah kemudian terpilihlah jalur terpendek antara sarang dan sumbermakanan.

Dalam algoritma koloni semut terdapat beberapa parameter masukan sebagai inisialisasi awal untuk melakukan proses optimasi. Beberapa parameter tersebut adalah:

- N : Banyak titik
- $\tau(i, j)$: Inisialisasi pheromone awal
- α : Parameter untuk mengendalikan tingkah kepentingan relatif dan jejak pheromone
- β : Parameter untuk mengendalikan tingkah kepentingan relatif visibilitas
- ρ : Evaporasi pheromone
- NC_{max} : Jumlah siklus maksimum, bersifat tetap selama algoritma dijalankan.

(Kusumadewi dan Purnomo, 2005)[5].

Dalam memilih rute perjalanan (titik yang akan dikunjungi selanjutnya), semut menggunakan persamaan peluang sebagai berikut :

$$P_{ij}(t) = \begin{cases} \frac{\tau_{ij}^\alpha(t)\eta_{ij}^\beta(t)}{\sum \tau_{ij}^\alpha(t)\eta_{ij}^\beta(t)} ; J \in U \\ 0; \text{lainnya} \end{cases}$$

dimana:

P_{ij} : Peluang transisi dari titik i ke j pada saat t

$\tau_{ij}(t)$: Intensitas jejak *pheromone* pada garis (i, j) pada saat (t)

$\eta_{ij}(t)$: Visibilitas node i dan j pada saat

$$t, \eta_{ij}(t) = \frac{1}{d(i, j)}$$

α : Parameter untuk mengendalikan tingkat kepentingan relative dari jejak *pheromone*

β : Parameter untuk mengendalikan tingkat kepentingan relatif dari visibilitas.

(Kusumadewi dan Purnomo, 2005)

Setelah semua semut melalui semua titik yang tersedia, proses berikutnya adalah melakukan perubahan nilai intensitas *pheromone* antar titik. Hal ini dilakukan karena jumlah semut yang melewati setiap garis berbeda-beda, semakin banyak semut yang melewati suatu garis maka intensitas *pheromone* akan semakin tinggi begitu juga sebaliknya. Berikut persamaan yang digunakan untuk menghitung nilai perubahan intensitas *pheromone* antar titik:

$$\tau_{ij}(t + N) = (1 - \rho)\tau_{ij}(t) + \rho\Delta\tau_{ij}(t, t + N)$$

dimana :

$\tau_{ij}(t)$: Intensitas jejak *pheromone* pada garis (i, j)

$\tau_{ij}(t + N)$: Intensitas jejak *pheromone* setelah semut menyelesaikan perjalanan dalam 1 siklus

$\Delta\tau_{ij}(t, t + N)$: Intensitas jejak *pheromone* yang ditinggalkan oleh semut k pada garis (i, j) pada interval t dan $t + 1$

L^k : panjang lintasan yang dilalui semut k

ρ : koefisien penguapan *pheromone*

N : banyak titik

Untuk mendiskusikan algoritma koloni semut, lingkungan yang akan digunakan adalah sebuah graph yang fully connected (setiap node memiliki baris ke node yang lain) dan bidirectional (setiap jalur bisa ditempuh bolak balik dua arah). Setiap garis memiliki bobot yang menunjukkan jarak antara dua titik yang dihubungkan oleh garis tersebut (mindaputra, 2009)[6].

Constrain

Constraint merupakan syarat yang tidak boleh dilanggar agar dapat mendapatkan solusi dari algoritma semut dalam kasus penjadwalan. Syarat ini bisa dibagi menjadi 2 kategori yaitu *soft constraint* dan *hard constraint*. *Soft constraint* berarti proses penjadwalan ini memiliki prioritas yang dilarang, tetapi masih boleh dilanggar karena masih bisa ditoleransi jika terjadi pelanggaran, sedangkan untuk *hard constraint* harus benar-benar terpenuhi sebab jika tidak dipenuhi maka solusi yang diinginkan tidak akan pernah bisa dicapai. Adapun *hard constraint* yang diterapkan dalam penjadwalan mata kuliah ini antara lain:

1. Dalam satu waktu tidak ada dosen yang mengajar di dua tempat.
2. Mata kuliah dengan semester sama, prodi yang sama, kelas yang sama tidak boleh dijadwalkan dalam hari yang sama dan waktu yang sama.
3. Ruang perkuliahan tidak boleh di pakai pada waktu yang bersamaan.
4. Mata kuliah 3 SKS dibagi jadi 2 SKS dan 1 SKS tidak boleh dijadwalkan di hari yang sama.
5. Mata kuliah 4 SKS dibagi menjadi 2 SKS dan 2 SKS tidak boleh dijadwalkan di hari yang sama.

Soft constraint ini tetap harus dipenuhi namun tidak wajib, jadi masih ada kemungkinan terjadi pelanggaran pada *soft constraint*. Hal ini terjadi apabila syarat *hard constraint* memaksakan kondisi tertentu sehingga mata kuliah yang memenuhi syarat *soft constraint* harus keluar dari persyaratan.



Soft constraint yang diterapkan dalam penjadwalan mata kuliah ini antara lain:

1. Mata kuliah semester awal dijadwalkan lebih pagi.
2. Pecahan mata kuliah dijadwalkan satu hari setelah pecahan sebelumnya.
3. Area mengajar dosen yang berkelanjutan ditempatkan di area yang berdekatan.
4. Dosen dapat memesan waktu mengajar tertentu yang diinginkan.

Berikut adalah tahapan algoritma semut menggunakan graph dapat dilihat pada gambar 2.8:

1. Dari sarang, semut berkeliling secara acak mencari makanan sambil mencatat jarak antara titik yang dilalui.



2. Ketika sampai di sumber makanan, total jarak dari tiap titik yang ditempuh dijumlahkan untuk mendapatkan jarak dari sarang ke makanan.

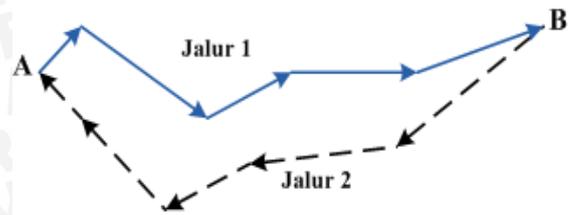
Gambar 2.1 Lintasan awal semut menuju tempat makanan

Keterangan Gambar 2.8:

A : Tempat awal koloni (sarang)
B : Tujuan koloni semut (makanan)

- Jalur 1 (biru) : Lintasan yang ditempuh semut 1
Jalur 2 (hitam) : Lintasan yang ditempuh semut 2

3. Ketika kembali ke sarang, sejumlah konsentrasi *pheromone* ditambahkan pada jalur yang ditempuh berdasarkan total jarak jalur tersebut. Makin kecil total jarak (atau makin optimal), maka makin banyak kadar *pheromone* yang dibubuhkan pada masing-masing garis pada jalur tersebut. Untuk lebih jelasnya dapat dilihat pada gambar 2.9.

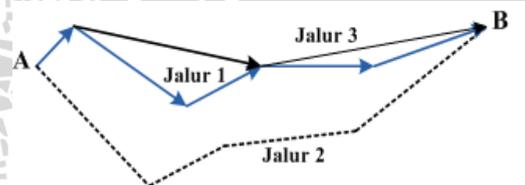


Gambar 2.2 Lintasan semut menuju sarang

Keterangan gambar 2.9:

- A : Sarang semut
B : Tempat ditemukannya makanan
Jalur 1 (biru) : Jalur yang ditempuh oleh semut 1 dengan pemberian kadar *pheromone* tinggi
Jalur 2 (hitam) : Jalur yang ditempuh oleh semut 1 dengan pemberian kadar *pheromone* tinggi

4. Untuk memilih garis mana yang harus dilalui berikutnya, digunakan sebuah rumus yang pada intinya menerapkan suatu fungsi *heuristic* untuk menghitung intensitas *pheromone* yang ditinggalkan pada suatu garis. Dapat dilihat pada gambar 2.10.

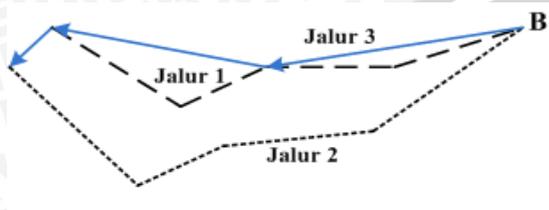


Gambar 2.3 Lintasan semut menuju makanan pada iterasi ke-2

Keterangan gambar 2.10:

- A : Sarang semut
B : Tempat ditemukannya makanan
Jalur 1 (biru) : Jalur yang ditempuh oleh semut 1 dengan pemberian kadar *pheromone* tinggi
Jalur 2 (hitam) : Jalur yang ditempuh oleh semut 1 dengan pemberian kadar *pheromone* tinggi
Jalur 3 : Jalur yang ditemukan semut 2

5. Pada iterasi berikutnya, garis-garis yang mengandung *pheromone* lebih tinggi ini akan cenderung dipilih sebagai garis yang harus ditempuh berikutnya. Akibatnya, lama-kelamaan akan terlihat **Jalur Optimal** pada graph, yaitu jalur-jalur yang dibentuk oleh garis dengan kadar *pheromone* yang tinggi, yang pada akhirnya akan dipilih oleh semua multi agen semut. Dapat dilihat pada gambar 2.11



Gambar 2.4 Lintasan semut menuju sarang pada iterasi ke-2

Keterangan gambar 2.11:

A : Sarang semut

B : Tempat ditemukannya makanan

Jalur 1 (biru) : Jalur yang ditempuh oleh semut 2 dengan pemberian kadar *pheromone* rendah

Jalur 2 : Jalur yang ditempuh

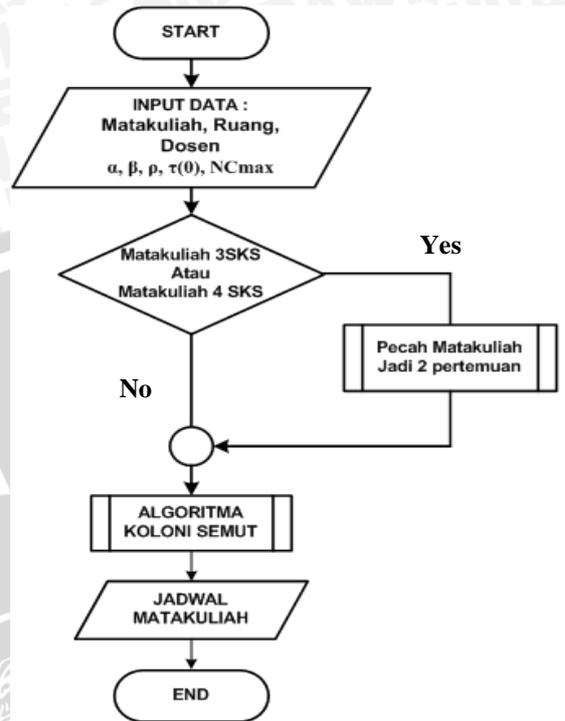
Jalur 3 (biru) : Jalur yang ditempuh oleh semut 2 dengan pemberian kadar *pheromone* yang tinggi

6. Berdasarkan rangkaian kejadian diatas, maka terpilihlah **jalur optimal** yaitu jalur ke-3 dengan menggunakan metode *heuristic* menggunakan algoritma koloni semut.

(Wardy, 2007)[7]

METODOLOGI DAN PERANCANGAN

Dalam penyelesaian masalah penjadwalan mata kuliah dengan algoritma koloni semut, urutan proses sistem ditunjukkan dalam flowchart dibawah ini.



Gambar Flowchart proses sistem penjadwalan kuliah

1. Langkah 1 : Input data mata kuliah, dosen pengajar dan ruangan yang akan digunakan. Dalam input mata kuliah terdapat jumlah SKS mata kuliah tersebut.

Input nilai parameter : α , β , ρ , $\tau(0)$ dan NC_{max} .

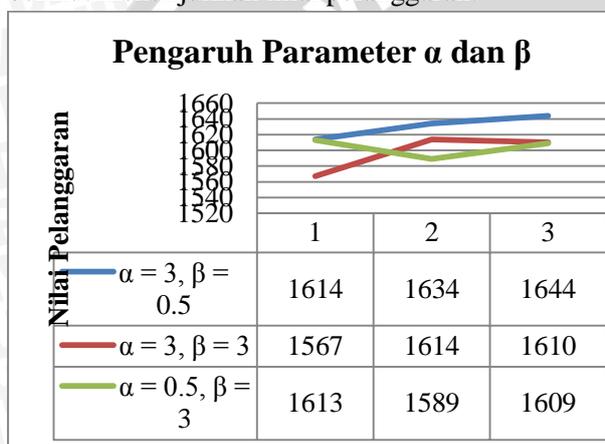
2. Langkah 2 : Dilakukan pengecekan terhadap jumlah SKS mata kuliah yang di inputkan. Mata kuliah 3 SKS atau 4 SKS menjadi dua pertemuan.

3. Langkah 3 : Proses algoritma koloni semut.

4. Langkah 4 : Jadwal mata kuliah yang dihasilkan.

Analisis hasil uji kinerja parameter α dan β

Dalam pengujian parameter α dan β dapat disimpulkan bahwa untuk nilai α dan β sedikit mempengaruhi nilai pelanggaran yang dihasilkan. Nilai α dan β yang terbaik adalah 3 dan 3 menghasilkan nilai terkecil 1567 dan memiliki rata-rata terkecil juga yaitu 1597 itu berarti nilai pelanggaran yang dihasilkan lebih kecil dibandingkan dengan nilai pelanggaran parameter yang lainnya. Apabila semakin kecil nilai pelanggaran yang dihasilkan maka semakin kecil pelanggaran jadwal yang dihasilkan oleh jadwal yang dibuat. Dapat dilihat pada gambar berikut menunjukkan nilai pelanggaran.



Untuk mengetahui masing-masing pelanggaran dapat dilihat pada tabel dibawah ini

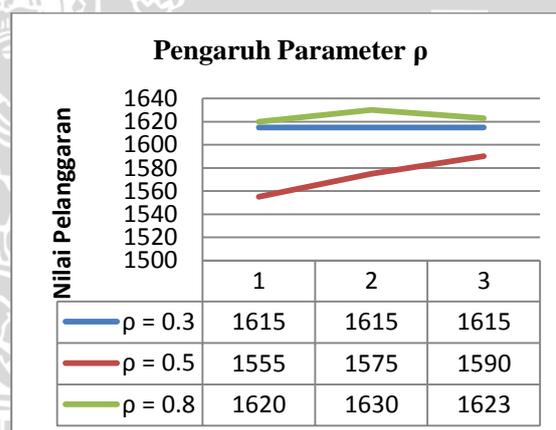
P	Nilai	Dosen	SKS	awal	Sebelah	area	waktu
p1	1555	0	0	720	209	0	3785
p2	1575	0	0	738	209	0	3766
p3	1590	0	0	786	171	7	3786

Pada percobaan yang pertama menunjukkan bahwa nilai pelanggaran total berjumlah 1567 dimana terjadi pelanggaran semester awal dijadwalkan lebih pagi sebanyak 750, pelanggaran pecahan SKS dijadwalkan selang satu hari setelah jadwal pada hari tersebut sebanyak 190 dan pelanggaran untuk dosen yang mengajar berkelanjutan area mengajar berdekatan memiliki nilai pelanggaran

berdekatan memiliki nilai pelanggaran sebanyak 0 dan waktu komputasi yang dibutuhkan untuk menyelesaikan proses penjadwalan sebanyak 15 generasi menempuh waktu 3853 detik.

Analisa hasil uji kinerja parameter ρ

Pada pengujian parameter α dan β telah di dapat nilai terbaik untuk α dan β adalah 3 dan 3. Untuk melengkapi agar nilai yang dihasilkan semakin baik maka selanjutnya menentukan nilai ρ yang sesuai untuk melengkapi tersebut adalah 0,5. Pada tabel 4.5 terdapat nilai terkecil yaitu 1555 dan memiliki rata-rata terkecil yaitu 1573 dibandingkan dengan nilai parameter yang lainnya. Pada gambar dibawah ini menunjukkan nilai pelanggaran.



Untuk lebih jelasnya dapat dilihat pada gambar di bawah ini dimana terdapat informasi tentang pelanggaran yang terjadi.

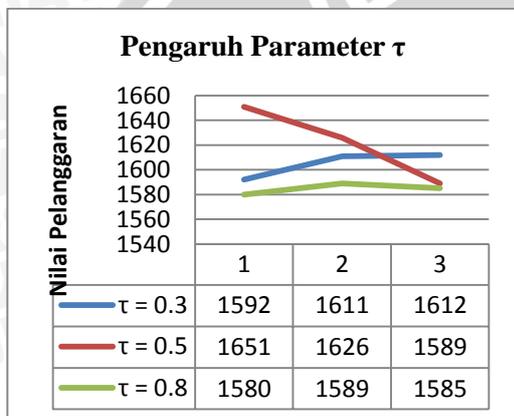
P	Nilai	Dosen	SKS	Awal	Sebelah	area	waktu
p1	1567	0	0	750	190	0	3853
p2	1614	0	0	816	171	0	4581
p3	1610	0	0	774	209	0	4462

Pada percobaan yang pertama menunjukkan bahwa nilai pelanggaran total berjumlah 1555 dimana terjadi pelanggaran semester awal dijadwalkan lebih pagi sebanyak 720, pelanggaran pecahan SKS dijadwalkan selang satu hari setelah jadwal pada hari tersebut sebanyak 209 dan pelanggaran untuk dosen yang mengajar berkelanjutan area mengajar berdekatan memiliki nilai pelanggaran

sebanyak 0 dan waktu komputasi yang dibutuhkan untuk menyelesaikan proses penjadwalan sebanyak 15 generasi menempuh waktu 3785 detik.

Analisa hasil uji kinerja parameter τ

Pada pengujian parameter α dan β telah di dapat nilai terbaik untuk α , β adalah 3,3 dan ρ 0,5. Untuk melengkapi agar dapat memberikan nilai yang baik, maka τ yang terbaik adalah 0,01. Pada tabel pengujian ρ terdapat nilai terkecil yaitu 1555 dan memiliki rata-rata terkecil yaitu 1573 dibandingkan dengan nilai parameter yang lainnya. Pada gambar dibawah ini menunjukkan nilai pelanggaran.



Untuk mengetahui masing-masing pelanggaran yang terjadi dapat dilihat pada tabel dibawah ini

Pada percobaan yang pertama menunjukkan

P	Nilai	Dosen	SKS	Awal	Sebelah	area	waktu
p1	1555	0	0	720	209	0	3785
p2	1575	0	0	738	209	0	3766
p3	1590	0	0	786	171	7	3786

bahwa nilai pelanggaran total berjumlah 1555 dimana terjadi pelanggaran semester awal dijadwalkan lebih pagi sebanyak 720, pelanggaran pecahan SKS dijadwalkan selang satu hari setelah jadwal pada hari tersebut sebanyak 209 dan pelanggaran untuk dosen yang mengajar berkelanjutan area mengajar berdekatan memiliki nilai pelanggaran sebanyak 0 dan waktu komputasi yang dibutuhkan untuk menyelesaikan proses

penjadwalan sebanyak 15 generasi menempuh waktu 3785 detik.

KESIMPULAN

Setelah melakukan penelitian maka disimpulkan :

1. Algoritma koloni semut dapat diterapkan dalam masalah Penjadwalan mata kuliah di Jurusan Matematika Fakultas MIPA Universitas Brawijaya Malang. Berdasarkan penelitian yang telah dilakukan, proses penjadwalan mata kuliah tidak terdapat pelanggaran untuk *hardconstrain*.
2. Pemilihan parameter awal algoritma sangat berpengaruh terhadap waktu komputasi. Parameter awal yang terbaik yaitu $\alpha = 3$ $\beta = 3$ $\rho = 0,5$ dan *pheromone* awal 0,01. Untuk 5 generasi waktu komputasi yang dibutuhkan sebesar 1539 s, 10 generasi membutuhkan waktu 2889 s, dan untuk 15 generasi membutuhkan waktu sebesar 3950 s.

DAFTAR PUSTAKA

Dorigo, Marco.,Caro, G.D dan Luca M. Gambardella. 1999. *Ant Algorithms for Discrete Optimization*. Artificial Life Volume 5,Belgian [3].

Fernandez, A dan E. Handoyo. 2008. *Analyzing Evolutionary Algorithim Method to Optimize Time Table System (Case Study: University Scheduling Time Table)*, Universitas Diponegoro. Semarang [4].

Kusumadewi, S. dan Purnomo,H. 2005. *Penyelesaian Masalah Optimasi dengan Teknik-teknik Heuristik*. Graha Ilmu.Yogyakarta.hal. 231-386[5].

Mindaputra, E. 2009.*Penggunaan algoritma ant colony system dalam traveling salesman problem (TSP) pada PT. EKA*

JAYA MOTOR. Jurusan Matematika
Universitas Diponegoro.Semarang [6].

Muttakhirah, I'ing. 2007. *Menentukan Jalur
Terpendek Menggunakan Algoritma
Semut*. Jurusan teknik informatika
Universitas Islam Indonesia.Yogyakarta
[2].

Pinedo, M. L., 1994. *Schedulling Theory,
Algorithms and Systems*. Springer. New
York. hal. 1-6 [1].

Wardy, I.S. 2007.*Penggunaan Graf dalam
Algoritma Semut untuk Melakukan
Optimisasi*.(online),
(<http://elesys.fsaintek.unair.ac.id>, diakses
15 Oktober 2011)[7].

Weise,T. 2009. *Global Optimization
Algorithms-Theory and
Application*.(online),([http://www.itweise.
de/projects/book.pdf](http://www.itweise.de/projects/book.pdf),Tanggal akses: 13
Oktober 2011).

Yang, F. C dan Yu H.H. 2008.*Ant Colony
Optimization Method for Time Window
Constrained Batching and Scheduling
Problem*.APIEMS.Bali.

Zhang, Jun , X. Hu dan J.H Zhong. 2006.
*Implementation of ant Colony
Optimization technique for job shop
scheduling problem*. Sun Yat-sen
University. PR China.

Zukhri,Z dan Shidiq, A.2004. *Algoritma Semut
pada penjadwalan produksi Jobshop*.
Universitas Islam Indonesia. Yogyakarta.

