

# Particle Swarm Optimization Untuk Sistem Informasi Penjadwalan *Resource* Di Perguruan Tinggi

Mansur<sup>a\*</sup>, Toni Prahasto<sup>b</sup>, Farikhin<sup>c</sup>

<sup>a</sup>Politeknik Negeri Bengkalis, Riau

<sup>b</sup>Program Studi Teknik Mesin, Fakultas Teknik, Universitas Diponegoro Semarang

<sup>c</sup>Jurusan Matematika, Fakultas Sains Matematika, Universitas Diponegoro Semarang

Naskah Diterima : 10 Januari 2014; Diterima Publikasi : 10 Februari 2014

---

## Abstract

Course timetabling management information system is a system focusing on data management resource and constraints in optimizing the use of available resources and avoid clashing in the process of making class timetabling, so that the resulting information will be effective. Results of such information may assist the college in planning the use of space, and to develop programs of study. Data resource and constraints are implemented using timetabling information system with PSO algorithm. The results of the data analysis using the PSO algorithm by combining six hard constraint and two soft constraints cannot produce an optimal solution, because there are still clashing lecturer-timeslot (soft1), but without combining both soft constraints can produce maximum solutions in the use of the room, where the solution with the best fitness value (0.333), c1(2,0), c2(2,0), w(0,2), and maximum of 10 iterations of the desired solution. The final results are timetabling management information system for resource utilization that generates class timetable information and use of the room.

**Keywords :** Management information system; PSO algorithm; Resource; Constraints; Course timetabling; Higher education

## Abstrak

Sistem informasi manajemen penjadwalan kuliah merupakan suatu sistem berfokus pada pengelolaan data *resource* dan *constraints* dalam upaya memaksimalkan penggunaan *resource* yang tersedia dan terhindar dari bentrok, sehingga informasi yang dihasilkan efektif. Hasil dari informasi tersebut dapat membantu perguruan tinggi dalam merencanakan penggunaan ruangan, dan melakukan pengembangan program studi. Data *resource* dan *constraint* diproses menggunakan algoritma PSO versus metoda klasik, serta menggunakan *software* FET dalam mengimplementasikan masalah penjadwalan kuliah secara otomatis. Hasil analisa data penjadwalan kuliah menggunakan algoritma PSO dengan menggabungkan enam *hard* dan dua *soft constraint* belum dapat menghasilkan solusi yang maksimal, sebab masih ada bentrok yang terjadi pada dosen-*timeslot* (*soft1*), namun tanpa menggabungkan *soft constraint* dapat menghasilkan solusi maksimal dalam penggunaan ruangan, dimana solusi terbaik dengan nilai *fitness* (0,333), parameter c1 (2,0), c2 (2,0), w (0,2), dan maksimal iterasi 10 dari solusi yang diinginkan. Hasil akhir penelitian adalah sistem informasi manajemen penjadwalan kuliah berbasis intranet dan desktop untuk pemanfaatan *resource* yang menghasilkan informasi jadwal kuliah dan penggunaan ruangan pada perguruan tinggi.

**Kata kunci :** Algoritma PSO; Pemrograman integer; Penjadwalan kuliah; *Resource*; *Constraints*; Perguruan tinggi

---

## 1. Pendahuluan

Sistem informasi manajemen penjadwalan kuliah pada merupakan upaya untuk meningkatkan kinerja akademik dalam mengelola, memproses data *resource* dan *constraint* yang tersedia. Pengelolaan tersebut bertujuan untuk menghasilkan informasi berkualitas dan efektif (Demirdjian, 2003), sehingga mampu memberikan informasi dalam pengambilan keputusan untuk mencapai tujuan perguruan tinggi (Kenneth dan Jane, 2010).

Penjadwalan adalah suatu alokasi, berfokus pada *constraints*, dan *resource* untuk *object* yang ditempatkan dalam ruang dan waktu (Pongcharoen et

al., 2008). Penjadwalan di perguruan tinggi merupakan masalah *nested partitioning* dan kompleksitas, seperti jadwal kuliah dan ujian akhir (Chu et al., 2006). Kompleksitas terjadi ketika jumlah mahasiswa dan pertemuan matakuliah yang harus dijadwalkan sangat banyak hingga mencapai ratusan bahkan ribuan. Dalam proses penjadwalan kuliah kita tidak hanya memperhatikan terjadinya bentrok pada level kelas dan dosen tetapi juga harus memperhatikan jadwal pertemuan semua mahasiswa agar tidak bentrok, sehingga jadwal yang dihasilkan dapat menjadi efisien dengan preferensi yang fleksibel (Shiau, 2011).

---

\*) Penulis korespondensi: mansur.polbeng82@gmail.com

Penjadwalan kuliah pada perguruan tinggi merupakan masalah pencarian yang melibatkan dan menugaskan satu set *timeslot* dalam meningkatkan kepuasan untuk mengatasi *constraint* yang merupakan peranan penting dalam menghasilkan jadwal yang optimal (Kano dan Chen, 2013).

Masalah tersebut secara umum memiliki karakteristik yang tidak jauh berbeda dengan perguruan tinggi lainnya, dimana jadwal kuliah selalu berkaitan dengan *timeslot*, ruangan, mata kuliah, mahasiswa, dosen dan *constraint*, dimana setiap tahun ajaran terdiri dari dua semester yaitu ganjil dan genap (Irene *et al.*, 2009).

Algoritma PSO berfokus pada penyelesaian masalah optimasi dalam pencarian ruang untuk mendapatkan solusi (Kennedy dan Eberhart, 1995). Algoritma PSO dapat menyelesaikan masalah penjadwalan kuliah dengan meminimalkan kesenjangan waktu dan memaksimalkan pemanfaatan *resource* dan *constraints* dalam penggunaan ruangan yang efektif (Teoh *et al.*, 2013).

Tujuan dari penelitian adalah merancang sistem informasi manajemen penjadwalan kuliah untuk pemanfaatan *resource* pada perguruan tinggi menggunakan PSO dengan memperhatikan *hard constraints* dan *soft constraints*.

## 2. Kerangka Teori

Penjadwalan akademik pada perguruan tinggi meliputi jadwal perkuliahan dan ujian akhir, jadwal perkuliahan bertujuan untuk meminimalkan kesenjangan antara waktu dosen dan mahasiswa, sementara jadwal ujian akhir untuk memaksimalkan kesenjangan waktu dan pengawas. Masalah tersebut dapat diterapkan dengan menggunakan metode *metaheuristic* seperti *tabu search (TS)*, *genetic algorithm (GA)*, *simulated annealing (SA)*, *particle swarm optimization (PSO)*, *fuzzy logic algorithm*, *ant colony optimization (ACO)*, and *hyper-heuristics*. Hasil dari penelitian dapat disimpulkan bahwa tidak ada algoritma tertentu yang lebih baik dalam memecahkan masalah sistem penjadwalan akademik dimana, setiap algoritma memiliki kekuatan yang berbeda-beda, namun dalam menyelesaikan masalah pada aspek pencarian ruang GA dan PSO mendapatkan hasil yang lebih baik. Teknik *metaheuristic* menunjukkan hasil yang menjanjikan dalam penyelesaian masalah penjadwalan akademik, dalam meningkatkan kualitas informasi dari aspek kecepatan secara komputasi, kelayakan dan optimalisasi (Teoh *et al.*, 2013).

Beberapa penelitian mengenai penjadwalan yang berfokus pada alokasi ruangan dan waktu dalam menyelesaikan masalah untuk mencapai tujuan yang telah ditentukan seperti, sistem penjadwalan sekolah tinggi di Yunani dengan menggunakan pendekatan algoritma PSO. Penerapan penjadwalan memperhatikan dua kategori yaitu; *hard constraints*

berhubungan dengan kepuasan dalam menghasilkan jadwal yang *flexible* dan *soft constraints* berhubungan kualitas sistem penjadwalan. Data yang digunakan adalah data pengajar, ruangan, hari, jam mata pelajaran, jam mengajar per hari, jam mengajar per kelas, total jam mengajar, total jam per kelas dan biaya. Semua data dari sekolah diasumsikan bahwa kapasitas ruangan lebih besar dari jumlah siswa. Hasil sistem dapat diterapkan secara efektif dan *flexible* dalam meningkatkan kinerja dan pemanfaatan *resource* (Tassopoulos dan Beligiannis, 2012).

Penjadwalan sekolah tinggi di Yunani dengan pendekatan GA. Data yang digunakan adalah *teachers, the courses, the classes and the time slots*. Hasil simulasi menunjukkan bahwa GA memberikan kualitas jadwal yang lebih baik dan *flexible* (Beligiannis *et al.*, 2009). Membangun sistem penjadwalan menggunakan variabel acak melalui teknik algoritma yang berfokus pada penerapan aplikasi *school timetable builder (STB)*. Data yang dibutuhkan terdiri dari guru, ruangan, dan waktu periode. Aplikasi STB dibuat menggunakan visual basic 6.0 yang berjalan pada microsoft windows. Penelitian ini menghasilkan aplikasi yang bekerja dengan variabel acak dan *output* sangat efektif (Panagiotakopoulos, 2008).

Berbeda dengan sistem penjadwalan di universitas menggunakan *Hybrid particle swarm optimization (HPSO)* dengan 6 *hard constraint* dan 4 *soft constraint* untuk menghasilkan preferensi yang *flexible*. Algoritma HPSO diuji pada universitas di Taiwan. Hasil penelitian ini menunjukkan bahwa algoritma HPSO dapat menyelesaikan masalah pemanfaatan *resource* dan *constraint* dalam menghasilkan jadwal kuliah yang kompleksitas dengan memperhatikan preferensi instruktur dan mahasiswa untuk mendapatkan solusi yang optimal (Shiau, 2011).

Dalam pemecahan masalah penjadwalan untuk mengoptimalkan penggunaan *resource* dan *constraint* tidak mudah sehingga komunitas peneliti AI (*artificial intelligence*) dalam penelitian dibidang *timetable* menemukan beberapa variasi dalam memecahkan masalah tersebut dengan menggunakan berbagai teknik. Teknik tersebut dapat dibagi menjadi empat tipe yaitu (Norberciak, 2009) :

- a. Metode sekuensial, metode ini menggunakan teknik heuristik dan menetapkan kegiatan secara berurutan kedalam periode waktu yang telah ditentukan (*timeslot*), sehingga kegiatan yang ditempatkan pada *timeslot* tidak bentrok dengan kegiatan yang lain. Untuk mengatasinya kegiatan yang paling sulit dalam pembuatan jadwal akan diletakkan pada *timeslot* pertama.
- b. Metode cluster, pada metode ini kegiatan dikumpulkan kedalam kelompok dimana setiap dua kegiatan dalam kelompok tersebut tidak bertentangan satu sama lainnya. Kelemahan

pendekatan ini pada pembentukan kelompok dari awal secara tetap sehingga menghasilkan kualitas *timetable* yang kurang baik.

- c. Metode berbasis kendala, pada metode ini *timetable* dimodelkan sebagai satu set variable domain berupa kegiatan yang memiliki keterbatasan *resource* yang harus ditempatkan pada *timeslot* untuk memenuhi sejumlah kendala yang ada.
- d. Metode metaheuristik, variasi metode metaheuristik dalam menyelesaikan masalah *timetable* yaitu: SA, tabu search, PSO, evolutionary algorithm, dan pendekatan *hybrid* lainnya. Metode tersebut bertujuan untuk menemukan strategi dalam proses pencarian dalam meningkatkan solusi optimal dan menghindari lokal optimal.

Penjadwalan kuliah merupakan bagian masalah *nested partitioning* (Cheu *et al.*, 2006; Staereling, 2012). Penjadwalan kuliah memiliki arti sebagai durasi waktu kerja yang dibutuhkan untuk melakukan aktivitas dalam kegiatan pembelajaran. Masalah tersebut berkaitan dengan *resource* dan *constraint* yang sudah ditentukan dalam proses perkuliahan yang membutuhkan alokasi waktu untuk menghasilkan solusi yaitu jadwal kuliah (Burke dan Rudova, 2006).

Penjadwalan kuliah juga merupakan masalah yang kompleksitas, terutama ketika jumlah mahasiswa dan pertemuan matakuliah yang harus dijadwalkan sangat banyak hingga mencapai ratusan bahkan ribuan, dimana hal tersebut menjadi semakin kompleks ketika proses penjadwalan tidak hanya memperhatikan terjadinya bentrok pada level kelas dan dosen tetapi juga harus memperhatikan jadwal pertemuan semua mahasiswa agar tidak bentrok. Untuk pembuatan jadwal pertemuan seperti ini akan mengalami kesulitan dalam mengatur *resource* dan *constraint* yang ada, sehingga sulit mendapatkan hasil yang optimal. Jadwal pertemuan terdiri dari satu set program yang akan dijadwalkan pada periode waktu dan jumlah hari serta memperhatikan *constraints* yang ada. Berikut *hard constraints* dan *soft constraints* dalam proses pembuatan jadwal kuliah (Shiau, 2011) yaitu:

a. *Hard constraints*

- 1. Setiap instruktur hanya dapat mengajar satu kuliah pada satu waktu.
- 2. Setiap perkuliahan hanya dapat diajarkan oleh satu instruktur dalam waktu yang sama.
- 3. Mahasiswa hanya dapat menghadiri satu perkuliahan pada satu periode.
- 4. Setiap ruang kelas tidak dapat digunakan lebih dari satu perkuliahan pada waktu yang sama.
- 5. Periode waktu tertentu tidak dapat dijadwalkan untuk kegiatan diluar akademik seperti waktu makan siang dan olahraga.

- 6. Beberapa matakuliah tertentu harus dijadwalkan dalam ruangan tertentu seperti laboratorium komputer.

b. *Soft constraints*

- 1. Instruktur dan mahasiswa dapat menunjukkan preferensi bersama dengan hari dan periode waktu yang disukai berdasarkan *timeslot* yang tersedia.
- 2. Instruktur dapat memilih jumlah jam tidak mengajar pada periode waktu tertentu.
- 3. Perkuliahan dalam waktu tiga jam harus dijadwalkan pada periode waktu berturut-turut dan tidak dapat dipisahkan.
- 4. Meminimalkan gerakan mahasiswa untuk melakukan perpindahan ruangan.

Gambar 1 menunjukkan representasi dari jadwal perkuliahan dalam satu set pertemuan dalam lima hari dengan 40 periode.

	Mon	Tue	Wed	Thu	Fri	...	Mon	Tue	Wed	Thu	Fri
1	9	17	25	33	...	$40(m-1)+1$	...	...	...	...	...
2	10	18	26	34	...	$40(m-1)+2$	...	...	...	...	...
3	11	19	27	35	...	$40(m-1)+3$	...	...	...	...	...
4	12	20	28	36	...	$40(m-1)+4$	...	...	...	...	...
5	13	21	29	37	...	...	...	...	...	...	...
6	14	22	30	38	...	...	...	...	...	...	...
7	15	23	31	39	...	...	...	...	...	...	$40m-1$
8	16	24	32	40	...	...	...	...	...	...	$40m$

Gambar 1. Representasi jadwal perkuliahan (Shiau, 2011)

Menurut Irene *et al.*, (2009) dalam penerapan penjadwalan kuliah pada Universitas ada beberapa *hard constraints* dan *soft constraints* yang digunakan untuk mendapatkan hasil jadwal yang lebih baik. Adapun *constraint* yang digunakan yaitu sebagai berikut :

a. *Hard constraints* :

- 1. Kendala bentrok waktu dosen; dosen tidak boleh mengajar lebih dari satu matakuliah dalam *timeslot* yang sama.
- 2. Kendala bentrok waktu *group* mahasiswa; satu *group* mahasiswa tidak boleh ditugaskan lebih dari satu matakuliah dalam *timeslot* yang sama.
- 3. Kendala bentrok ruangan kelas; satu ruangan tidak boleh ditugaskan lebih dari satu matakuliah untuk *timeslot* yang sama.
- 4. Kendala kapasitas ruangan kelas; jumlah mahasiswa yang mengikuti pembelajaran pada ruangan kelas tidak boleh melebihi kapasitas ruangan.
- 5. Kendala ruangan kelas dan *domain-timeslot*; ruangan kelas atau *timeslot* yang ditugaskan untuk matakuliah harus berada pada domain yang ada.
- 6. Kendala *timeslot*; untuk kegiatan bukan akademik seperti ekstra kurikulum dan jam

makan siang disediakan timeslot tertentu sehingga mereka tidak bersedia untuk melakukan proses perkuliahan.

b. *Soft constraints* :

1. Dosen dapat menentukan jadwal dengan preferensi matakuliah dan periode waktu berdasarkan *timeslot* yang tersedia.
2. Menentukan jadwal dengan preferensi matakuliah terhadap ruang kelas atau laboratorium yang tersedia.

Pada Tabel 1, menunjukkan representasi jadwal kuliah dalam satu set pertemuan dalam lima hari dengan 43 periode.

Tabel 1. Jadwal kuliah dalam satu minggu (Irene *et al.*, 2009).

Time	Mond	Tuesd	Wed	Thurs	Frid
08:00 - 08:50	1	2	3	4	5
09:00 - 09:50	6	7	8	9	10
10:00 - 10:50	11	12	13	14	15
11:00 - 11:50	16	17	18	19	20
12:00 - 12:50	21	22	23	24	
13:00 - 13:50					
14:00 - 14:50	25	26		27	
15:00 - 15:50	28	29		30	31
16:00 - 16:50	32	33		34	35
17:00 - 17:50	36	47		38	39
18:00 - 18:50	40	41		42	43

PSO merupakan salah satu teknik dasar dari *swarm intelligence system* untuk menyelesaikan masalah optimasi dalam pencarian ruang sebagai suatu solusi. PSO pertama kali diusulkan oleh James Kennedy dan Eberhart (1995), dirancang untuk mensimulasikan burung dalam proses pencarian makanan. *Swarm intelligence system* melakukan penyebaran kecerdasan yang inovatif dalam menyelesaikan masalah optimasi dengan mengambil inspirasi dari contoh biologis, seperti fenomena kelompok (*swarm*) pada hewan, dimana setiap kelompok memiliki perilaku individu dalam melakukan tindakan bersama untuk mencapai tujuan yang sama.

*Swarm* biasa disebut agen, setiap agen bertindak berdasarkan aturan lokal meskipun tidak ada pusat kendali untuk setiap agen dalam melakukan reaksi, sehingga terbentuknya suatu kecerdasan (*intelligence*) global tanpa disadari oleh setiap agen dalam suatu kawanan untuk mencapai tujuan. Ada beberapa *properties* yang harus diperhatikan dalam proses *swarm intelligence system* seperti kesatuan (*unity*), toleransi kesalahan (*fault tolerance*), perilaku berbasis aturan (*rule-based behavior*), otonomi (*autonomy*), skalabilitas (*scalability*), adaptasi (*adaptation*), kecepatan (*speed*), modularitas (*modularity*), dan memiliki sifat secara parallel (Sumathi dan Surekha, 2010).

PSO merupakan salah satu metode metaheuristik yang pencarian solusi berdasarkan populasi dari sekelompok burung atau ikan, dimana setiap populasi memiliki individu yang dapat mempengaruhi individu lainnya. Individu tersebut disebut juga sebagai partikel yang diperlakukan seperti sebuah titik pada suatu dimensi ruang waktu tertentu (Santoso dan Willy, 2011). Pada algoritma PSO pencarian solusi dilakukan secara acak dari suatu populasi, dimana setiap partikel berkaitan dengan posisi dan kecepatan dalam melakukan pencarian baru secara dinamis berdasarkan perilaku mereka. Setiap partikel memiliki nilai *fitness* yang harus dievaluasi untuk setiap generasi berdasarkan *local best* (*pbest*) dan *global best* (*gbest*) yang merupakan pengalaman dari setiap partikel dalam menghasilkan solusi terbaik. Pengalaman tersebut dapat digunakan sebagai parameter *weight inertia* dalam menentukan pengaruh kecepatan sebelumnya dengan kecepatan baru (Shiau, 2011).

Nilai *weight inertia* dapat memperbaiki kecepatan partikel dalam menyeimbangkan proses pencarian global dan lokal (Eberhart dan Shi, 2001; Grosan, *et al.*, 2006; Bratton dan Kennedy, 2006). Pada kasus penjadwalan ini, partikel diwakili oleh slot-slot dalam satu set jadwal yang terdiri dari beberapa kelas yang merupakan satu partikel (Ariani *et al.*, 2010; Rachman, *et al.*, 2012; Rifa'i dan Kurniawati, 2013).

Adapun parameter yang digunakan dalam proses algoritma PSO adalah sebagai berikut (Sumathi dan Surekha, 2010):

- a. Jumlah partikel (*Number of particles*) merupakan faktor yang dianggap sangat penting dalam melakukan penyelesaian masalah.
- b. Bobot Inersia (*Inertia Weight*), memainkan peran yang sangat penting dalam kecepatan partikel dari algoritma PSO.
- c. Faktor pembelajaran (*Learning factors*). Parameter *c1* merupakan pengakuan koefisien, sedangkan *c2* adalah komponen social. Hal ini tidak terlalu penting untuk perilaku konvergensi dari algoritma PSO
- d. Rentang dan dimensi partikel (*Range and dimension of particles*). Dimensi partikel dan rentang ditentukan berdasarkan masalah yang dioptimalkan.
- e. Kecepatan (*Velocity*) merupakan perubahan maksimum pada setiap partikel, dapat diambil selama iterasi yang didefinisikan sebagai kecepatan maksimum.
- f. Menghentikan kondisi (*Stopping condition*) merupakan salah satu cara apabila kriteria yang dicari sudah tercapai.

Istilah yang digunakan dalam penerapan algoritma PSO sebagai berikut:

- a. *Swarm*: populasi dari sekawanan partikel
- b. *Particle*: individu pada suatu *swarm*. Setiap partikel mempresentasikan suatu solusi dari permasalahan yang diselesaikan.

- c. *Pbest*: suatu partikel yang menunjukkan posisi terbaik.
- d. *Gbest*: posisi terbaik dari seluruh partikel yang ada dalam suatu *swarm*.
- e. *Velocity*: kecepatan yang dimiliki oleh setiap partikel dalam menentukan arah perpindahan suatu partikel untuk memperbaiki posisi semula.
- f. *c1* dan *c2*: *c1* merupakan konstanta pembelajaran kognitif, dan *c2* konstanta pembelajaran sosial.

Ada 4 tahapan algoritma PSO yang digunakan untuk menyelesaikan masalah penjadwalan kuliah yaitu (Shiau, 2011):

1. *Initialization*

Inisialisasi suatu populasi dengan posisi partikel, dan kecepatan dari sekumpulan partikel yang dibangkitkan secara random dengan menggunakan batas atas ( ) dan batas bawah ( ). Dapat dirumuskan dalam persamaan 1, dan 2 yaitu (Rachman *et al.*, 2012):

$$= + ( - ) \quad (1)$$

$$= + ( - ) \quad (2)$$

Posisi partikel dan kecepatan dapat di tampilkan dalam bentuk vektor dimana *n* dimensi vektor yang direpresentasikan dengan notasi partikel ke *i* pada waktu ke *k*. Proses inisialisasi dari kumpulan partikel dapat terdistribusi secara random pada rancangan ruang (*design space*).

2. *Evaluation* fungsi *fitness*

Fungsi *fitness* bertujuan untuk mengukur sejauh mana partikel dalam mendapatkan solusi yang baik dalam memenuhi *constraint*. Setiap partikel dievaluasi dengan menggunakan fungsi *fitness* untuk mengetahui partikel yang memiliki nilai *fitness* yang terbaik (Ariani, *et al.*, 2010; Suyanto, 2010). *fitness* terbaik adalah nilai *fitness* yang memiliki jumlah pelanggaran yang kecil dalam menghasilkan solusi. Untuk setiap pelanggaran yang terjadi akan diberikan nilai 1. Agar tidak terjadi nilai *fitness* yang tak terhingga maka jumlah total semua pelanggaran ditambah 1 (Santoso dan Willy, 2011).

3. *Update*

Hitung kecepatan dari setiap partikel dan pindahkan posisi partikel yang didapat dari nilai local terbaik dan global terbaik. Ada tiga parameter yang perlu diperhatikan dalam pencarian pada algoritma PSO yaitu *w*, *c1*, *c2* yang diformulasikan dalam persamaan 3 (Rachman, *et al.*, 2012; Rifa'i dan Kurniawati, 2013) yaitu:

$$= * + * * - + * \quad (3)$$

$$= + \quad (4)$$

Keterangan :

*w* =Faktor inersia (*inertia weight*) batasan nilai antara (0,14 - 1,14).

=Kecepatan partikel.

=Posisi partikel

*rnd* =Parameter *random* yang terdistribusi dengan batasan nilai [0 – 1].

*c1,c2*=Konstanta faktor pembelajaran untuk kemampuan partikel dan pengaruh sosial dari kawanannya. *c1* = (1,5 – 2.0) dan *c2* = (2 - 2,5).

= *Local best*

= *Global best*

*i* = Partikel

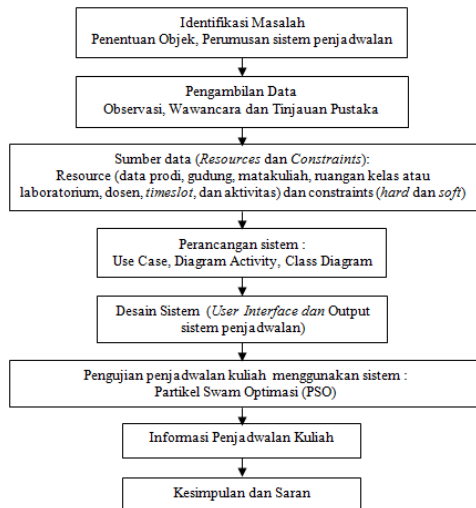
4. *Termination*

Algoritma PSO berakhir apabila kriteria yang dicari sudah tercapai, jika belum maka lanjutkan ke iterasi berikut dan tahapan ke 2, begitu seterusnya.

3. **Metodologi**

Prosedur dalam penelitian ini diawali dengan melakukan identifikasi masalah, penentuan program studi, menentukan permasalahan dari sistem penjadwalan serta mempelajari konsep-konsep mengenai penjadwalan, analisa kebutuhan sistem, penerapan menggunakan algoritma PSO sebagai pengujian sistem. Penentuan algoritma PSO yang relevan merujuk pada (Shiau, 2011 dan Irene *et al.*, 2009).

Pengambilan data dilakukan dengan Metode observasi, wawancara, dan studi pustaka. Pengambilan sampel data berdasarkan karakteristik yang ditentukan sesuai dengan masalah penelitian. Dalam hal ini data yang digunakan adalah data skunder diambil dari perguruan tinggi Politeknik Negeri Bengkalis, untuk semester ganjil tahun ajaran 2012/2013 dengan tujuh program studi yaitu teknik kapal, teknik mesin, teknik sipil, teknik elektro, teknik informatika, administrasi bisnis dan bahasa inggris bisnis. Data yang dibutuhkan berupa data *resources* (timeslot, dosen, mahasiswa, matakuliah, gedung, ruangan, dan program studi) dan data *soft constraint* (Oates, 2007).



Gambar 2. Prosedur Penelitian

Untuk menyusun penjadwalan kuliah pada perguruan tinggi Politeknik Negeri Bengkalis ada dua *constraints* yang harus diperhatikan yaitu :

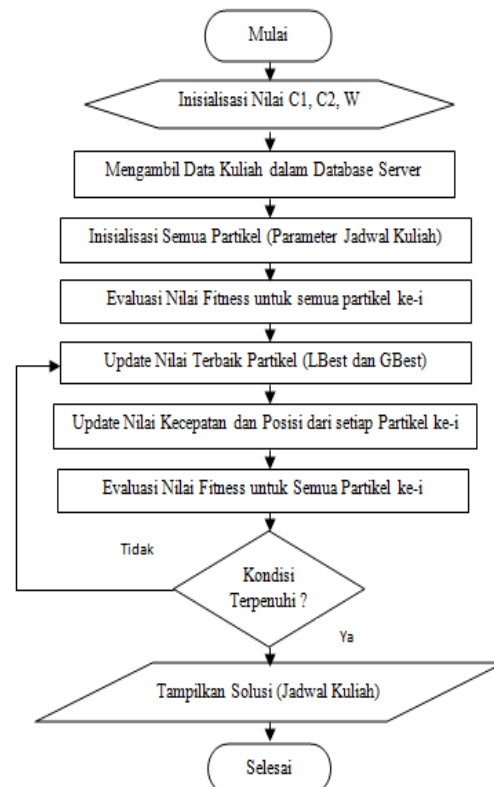
a. *Hard constraints*

1. Dosen tidak boleh mengajar lebih dari satu matakuliah dalam *timeslot* yang sama (Irene *et al.*, 2009).
2. Satu *group* kelas mahasiswa tidak boleh ditugas lebih dari satu matakuliah dalam *timeslot* yang sama (Irene *et al.*, 2009).
3. Satu ruangan tidak boleh ditugaskan lebih dari satu matakuliah untuk *timeslot* yang sama (Irene *et al.*, 2009).
4. Jumlah mahasiswa yang mengikuti pembelajaran pada ruangan tidak boleh melebihi kapasitas dari ruangan kelas (Irene *et al.*, 2009).
5. Beberapa matakuliah tertentu harus dijadwalkan dalam ruangan tertentu seperti laboratorium komputer (Shiau, 2011).
6. Pada hari jum'at, untuk periode 11:40 s/d 14:20 adalah jam istirahat sholat.

b. *Soft constraints*

1. Dosen dapat menunjukkan preferensi dengan hari dan periode waktu yang disukai berdasarkan *timeslot* yang tersedia (Shiau, 2011).
2. Ruangan dengan preferensi hari dan periode jam dalam kesediaan melaksanakan proses perkuliahan.

Algoritma PSO secara umum terdiri dari 4 tahapan yaitu *initialization*, *evaluation*, *update* dan *termination*. Dimana keempat tahapan tersebut sudah dilalui maka dilakukan pengecekan apakah hasil yang diperoleh sudah optimal atau belum, jika belum maka proses akan berulang sampai mendapatkan hasil yang optimal, jika proses sudah optimal maka proses akan berhenti. Berikut *flowchart* dari algoritma PSO, dapat dilihat pada Gambar 3.

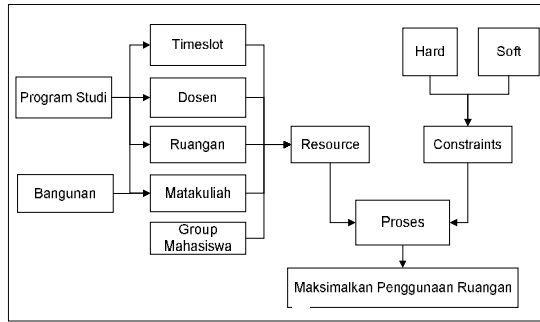
Gambar 3. *Flowchart* algoritma PSO untuk penjadwalan kuliah

Pada kasus penjadwalan kuliah ini, partikel adalah satu set jadwal kuliah yang terdiri dari beberapa kelas (Rachman, *et al.*, 2012; Rifa'i dan Kurniawati, 2013).

### 3. Hasil dan Pembahasan

Sistem informasi penjadwalan kuliah dengan algoritma PSO dibangun berbasis web dan desktop dengan menggunakan bahasa pemrograman ASP.Net/VB.Net, dan SQLServer 2008 r2 Express sebagai database server. Penelitian menggunakan studi kasus pada perguruan tinggi Politeknik Negeri Bengkalis.

Sistem informasi penjadwalan kuliah merupakan sebuah sistem yang dapat mengelola data *resource* dan *constraint* (Gambar 4) yang tersedia pada perguruan tinggi untuk mengoptimalkan penggunaan waktu ruangan. Input dari sistem bersumber dari tiga bagian yaitu bagian akademik (BAK), program studi dan pimpinan, sedangkan data input yaitu data *resource* dan *constraint*.



Gambar 4. Komponen dalam membangun program penjadwalan kuliah

Hasil analisa data penjadwalan kuliah menggunakan algoritma PSO dengan menggabungkan *hard constraint* dan *soft constraint* belum dapat menghasilkan jadwal yang optimal sesuai dengan target yang diinginkan, disebabkan masih ada bentrok dosen-timeslot (*soft1*), namun tanpa menggabungkan *soft constraint* dapat menghasilkan jadwal yang optimal dengan nilai target yang telah ditentukan.

Sistem penjadwalan dengan algoritma PSO dilakukan guna untuk mengelola *resource* dan *constraint* yang tersedia, sehingga penggunaan *resource* dapat dimanfaatkan secara optimal oleh perguruan tinggi Politeknik Negeri Bengkalis. Input data bersumber dari tiga entitas yaitu bagian akademik sebagai administrator, sedangkan program studi dan pimpinan sebagai pengguna biasa. Data yang sudah diinput oleh BAK, selanjutnya dilakukan proses untuk pembuatan jadwal kuliah. Berikut algoritma PSO untuk penjadwalan kuliah dapat dilihat pada Gambar 5.

**Inisialisasi :**

1. Tentukan timeslot awal secara default (Inisialisasi Timeslot, ruang-waktu (RT), Ketersediaan Ruang (Default 'Y'), dan Inisialisasi Ketersediaan Dosen (Default 'Y')).
2. Tentukan parameter  $w$ ,  $c1$ ,  $c2$ , dan  $V_{mak}$ .
3. Untuk setiap partikel ke  $i$  dalam suatu populasi (swarm).
  - a. Tentukan Kombinasi dari suatu partikel (DMKRT) - Inisialisasi partikel secara acak kedalam RT.
  - b. Hitung *fitness* [i].
  - c. Hitung *velocity*  $V[i]$  secara random.
  - d. Update *Position* local best [i]
  - e. Mencari posisi global best dari suatu populasi.

**4 Proses PSO Algorithm ():**

- 1 Panggil fungsi `init_sqdb()`
- 2 Panggil fungsi `qinit_swarm()`
- 3 Do While Not Done
- 4 loop berhenti jika target terpenuhi atau max epoch tercapai
- 5 If  $Epoch < max\_epoch$  Then
- 6 cari global best
- 7 cek apakah pBest lbh baik dari pada gBest
- 8 If  $Math.Abs(target\_min - qhitung\_fitness(gBest))$

```

    < _
    Math.Abs(target_min - qhitung_fitness(gBest)) Then
    9 gBest = gBestTest
    10 End If
    11 If (Math.Abs(target_min - qhitung_fitness(gBest)) = 0)
        Then
    12 Done = True
    13 End If
    14 cari kecepatan (formula 3)
    15 update posisi (formula 4)
    16 Lakukan pengulangan iterasi sampai menemukan
        solusi yang terbaik.
    17 Else
    18 Iterasi berhenti dengan batas maksimal iterasi
    19 End If
    20 Loop
    21 Ambil satu partikel yang memiliki nilai fitness yang
        nilai maksimum untuk dijadikan sebagai pembuatan
        jadwal.
    22 Simpan kedalam tabel schedule.
    
```

Gambar 5. Algoritma PSO dalam sistem penjadwalan

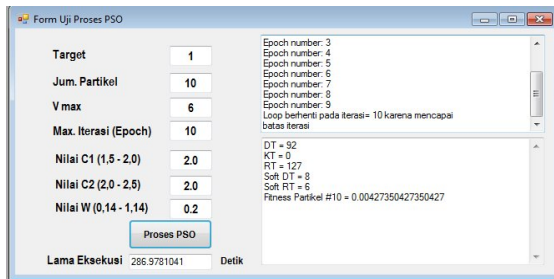
Pengujian sistem informasi penjadwalan kuliah menggunakan data *resource* dan *constraint* yang tersedia, kemudian dilanjutkan dengan proses pembuatan jadwal secara otomatis. Data rekapitulasi *resource* dapat dilihat pada Tabel 2. Sedangkan *constraint* yang digunakan dalam mengatur proses penjadwalan kuliah dapat dilihat pada bagian 3 (*hard* dan *soft constraint*).

Hasil analisa data *resource* menggunakan algoritma PSO dengan menggabungkan *hard constraint* dan *soft constraint* belum dapat menghasilkan jadwal yang optimal sesuai dengan target yang diinginkan, disebabkan masih ada bentrok dosen-timeslot (*soft1*) dengan nilai *fitness* terbaik (0,333), nilai parameter  $c1$  (2,0),  $c2$  (2,0),  $w$  (0,2), maksimal iterasi 10 dengan nilai target yang dicapai (Gambar 6), namun tanpa menggabungkan *soft constraint* dapat menghasilkan jadwal yang optimal dengan nilai target yang telah ditentukan (Gambar 7).

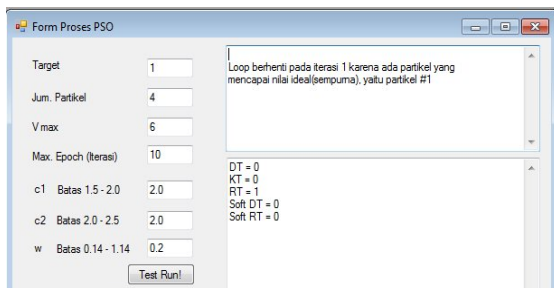
Tabel 2. *Resource* yang tersedia di perguruan tinggi.

No	Keterangan	Jumlah
1	Program studi : Teknik Informatika, Administrasi Bisnis, Bahasa Inggris, Teknik Kapal, Teknik Mesin, Teknik Elektro, dan Teknik Sipil.	7 Prodi
2	Hari Per Minggu	6 Hari
3	Jam Periode Per Hari	10 Periode
4	Total matakuliah semester ganjil 2012-2013 (SMT 1, 3 dan 5)	215 Matakuliah
5	Jumlah dosen tetap dan dosen luar biasa untuk proses pembelajaran	73 Orang
6	Jumlah mahasiswa Politeknik Negeri Bengkalis secara keseluruhan.	867 Orang
7	Jumlah gedung yang tersedia	6 Gedung
8	Jumlah ruangan kelas	20 Kelas
9	Jumlah ruangan laboratorium	34 Lab





Gambar 6. Proses penjadwalan kuliah menggunakan *soft constraint*



Gambar 7. Proses penjadwalan kuliah tanpa memperhatikan *soft constraint*

Pada Tabel 3 menunjukkan bahwa proses data *resource* dan *constraint* menggunakan algoritma PSO dengan jumlah partikel 10 dalam suatu populasi, dimana setiap partikel memiliki nilai *fitness* dari target yang telah ditentukan. Berdasarkan 10 partikel tersebut terdapat tiga partikel yang memiliki nilai *fitness* yang mendekati nilai target satu yaitu partikel ke 1,3,4,5,7 dan 9. Dari tiga partikel tersebut diambil partikel ke 9 untuk pembuatan jadwal kuliah pada Perguruan Tinggi Politeknik Negeri Bengkalis.

Tabel 3. Nilai *fitness* untuk setiap partikel.

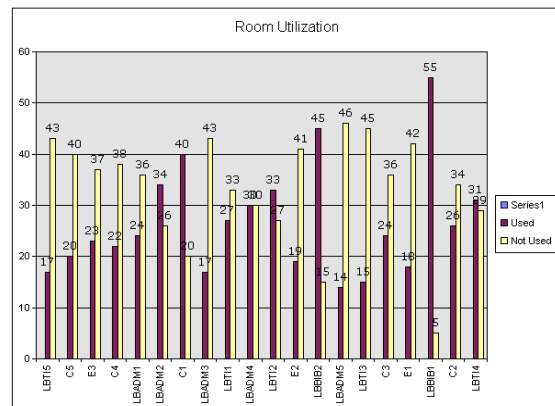
Partikel Ke-i	<i>Fitness</i>	Parameter PSO	Target	Waktu Proses
1	0.3333			
2	0.0047			
3	0.3333			
4	0.3333	W(0,2),		
5	0.3333	c1(2,0),	1	4 menit
6	0.0046	c2(2,0),		8 detik
7	0.3333	Iterasi (10)		
8	0.0097			
9	0.3333			
10	0.0060			

Dari *resource* yang ada pada partikel ke-9 dapat menghasilkan informasi berupa jadwal kuliah berdasarkan nama dosen, ruangan dan group mahasiswa, program studi dan pemanfaatan penggunaan ruangan (kelas dan laboratorium). Gambar 8 menunjukkan contoh informasi penjadwalan kuliah berdasarkan program studi. Informasi berisi tentang hari, jam mengajar, ruangan,

dosen, group mahasiswa dan matakuliah yang disajikan.

Program Studi	Mata Kuliah	Group lth.	Semester	Cursus	Hari	Mulai	Selesai	Dosen	Ruangan
Teknik Sipil	Gambar Teknik 1	A	1	4	RABU	09:30	11:20	Eko Dharma Putra, ST	Lab Perencanaan dan Pemukiman
Teknik Sipil	Pengenalan Komputer 1	A	1	4	JUMAT	09:30	11:20	Riba Supari, S.Si	Lab Perencanaan dan Pemukiman
Teknik Sipil	Konstruksi Aspal dan Perancah	A	3	4	RABU	09:30	11:20	Noerdin Basri, ST, IIT	Bengkel Batu, Beton dan Kayu
Teknik Sipil	Konstruksi Beton	A	3	4	SABTU	09:30	11:20	Alamsyah, ST, M.Eng	Bengkel Batu, Beton dan Kayu
Teknik Sipil	Konstruksi Jembatan	A	3	4	JUMAT	09:30	11:20	Rio Zembika, ST	Bengkel Batu, Beton dan Kayu
Teknik Sipil	Ilmu Usah Tani 1	A	1	4	SELASA	09:30	11:20	Noerdin Basri, ST, IIT	Lab Ilmu Usah Tani
Teknik Sipil	Laboratorium Uji Bahan 1	A	3	4	RABU	09:30	11:20	Alamsyah, ST, M.Eng	Lab Uji Bahan
Teknik Sipil	Alat Berat dar PTM	A	5	4	SELASA	09:30	11:20	Arif Fachri, SST	Lab Jaran Raya

Gambar 8. Jadwal kuliah berdasarkan program studi



Gambar 9. Pemanfaatan ruangan.

Gambar 9 dapat menunjukkan informasi pemanfaatan *resource* ruangan dengan menampilkan informasi mengenai penggunaan *timeslot* ruangan yang digunakan dalam proses perkuliahan. Informasi tersebut berupa total jumlah jam yang digunakan dan jumlah jam kosong pada setiap ruangan yang tersedia, sehingga dari informasi tersebut dapat membantu perguruan tinggi dalam merencanakan penggunaan ruangan, dan melakukan pengembangan program studi.

### 5. Kesimpulan

Sistem informasi manajemen penjadwalan kuliah dibangun menjadi dua yaitu berbasis web (*local*) dan desktop yang dapat diakses oleh pengguna, merupakan suatu sistem untuk dapat mengelola *resource* dan *constraints* yang tersedia. *Resource* tersebut diolah menggunakan algoritma PSO untuk membantu mengoptimalkan penggunaan ruangan dalam proses pembuatan jadwal kuliah secara otomatis.

Hasil analisa data *resource* dan *constraint* menggunakan algoritma PSO dengan memperhatikan *hard constraint* dan *soft constraint* belum dapat menghasilkan solusi yang optimal, karena masih terdapat bentrok dosen-*timeslot* (*soft1*), namun tanpa menggabungkan *soft constraint* dapat menghasilkan solusi yang optimal dalam penggunaan ruangan,



dimana solusi terbaik dengan nilai *fitness* (0,333), c1 (2,0), c2 (2,0), w (0,2), dan maksimal iterasi 10 dari nilai target yang dicapai.

Dari penelitian ini, masih perlu pengembangan terutama dalam hal penambahan algoritma PSO dengan metode metaheuristik yang lain, penambahan *soft constraint* seperti preferensi mahasiswa dalam menentukan pembuatan jadwal kuliah, memperhatikan jarak tempuh mahasiswa antar gedung, dan terintegrasinya sistem informasi akademik dengan sistem informasi penjadwalan kuliah sehingga hasil yang diperoleh dalam proses pembuatan jadwal kuliah menjadi lebih fleksibel dan optimal.

#### Daftar Pustaka

- Ariani, D., Fahriza, A. dan Prasetyaningrum, I., 2010. Optimasi penjadwalan mata kuliah di jurusan teknik informatika pens dengan menggunakan algoritma Particle Swarm Optimization (PSO), PENS-Surabaya.
- Beligiannis, G.N., Moschopoulos, C. and Likothanassis, S.D., 2009. A genetic algorithm approach to school timetabling, *Journal of the Operational Research Society*, 60(1): 23–42.
- Bratton, D. and Kennedy, J., 2006. Defining a Standard for Particle Swarm Optimization, *Proceedings of IEEE Swarm Intelligence Symposium*. 1-4244-0708-7, IEEE.
- Chu, C.S., Chen, T.Y. and Ho, H.J., 2006. Timetable Scheduling Using Particle Swarm Optimization, *Proceedings of the First International Conference on Innovative Computing, Information and Control*, 0-7695-2616-0/06, IEEE.
- Demirdjian, Z.S., 2003. Marketing research and information systems: The unholy separation of the Siamese twins, *Journal of American Academy of Business*, Cambridge; Sep 2003.
- Eberhart, C.R. and Shi, Y., 2001. Particle Swarm Optimization, *Developments, Applications and Resources*. 0-7803-6657-3/01, IEEE.
- Grosan, C., Abraham, A., and Chis, M., 2006. Swarm Intelligence in Data Mining, *Studies in Computational Intelligence*, Romania
- Irene, F.S.H., Deris, S. and Hashim, M.Z.S., 2009. Incorporating Of Constraint-Based Reasoning Into Particle Swarm Optimization For University Timetabling Problem, *ISSR Journals*, 1(1): 13–22.
- Kanoh, H., and Chen, S., 2013. Particle Swarm Optimization with Transition Probability for Timetabling Problems, *LNCS*, Springer.
- Kennedy, J. and Eberhart, R., 1995. Particle Swarm Optimization, *Purdue School of Engineering and Technology*, 0-7803-2768-3/95, IEEE.
- Kenneth, C.L. and Jane, P.L., 2010. Management Information System, *Managing The Digital Firm Eleventh Edition*. Pearson Prentice Hall.
- Norberciak, M., 2006. Universal Method for Timetable Construction based on Evolutionary Approach, *World Academy of Science, Engineering and Technology*.
- Oates, J.B., 2007. *Researching Information Systems and Computing*, SAGE Publication.
- Panagiotakopoulos, T.C., 2008. Building Teaching Timetables Using Random Variables: Algorithms and Techniques, *Innovations and Advanced Techniques in Systems, Computing Sciences and Software Engineering*, 131–136, Springer.
- Pongcharoen, P., Promtet, W., Yenradee, P. and Hicks, C., 2008. Stochastic Optimisation Timetabling Tool for university course scheduling, *Int. J. Production Economics*: 112: 903–918, Elsevier.
- Rachman, A.R., Syarif, D. and Sari, P.R., 2012. Analisa dan penerapan metode particle swarm optimization pada optimasi penjadwalan kuliah, *Jurnal Teknik Informatika*, Vol 1.
- Rifa'i, I.H.Q. dan Kurniawati, A., 2013. Analisis dan penerapan algoritma particle swarm optimization pada optimasi penjadwalan matakuliah di jurusan Teknik Informatika Universitas Trunojoyo Madura, *Jurnal Teknik Informatika*, 1(1): 1–4.
- Santoso, B. dan Willy, P., 2011. *Metoda Metaheuristik Konsep dan Implementasi*, Guna Widya.
- Shiau, F.D., 2011. A hybrid particle swarm optimization for a university course scheduling problem with flexible preferences, *Expert Systems with Applications*, 38: 235–248, Elsevier.
- Staereling, V.H.V., 2012. *School timetabling in theory and practice*, VU University, Amsterdam.
- Sumathi, S. and Surekha, P., 2010, *Computational Intelligence Paradigms Theory and Applications Using Matlab*, Taylor and Francis Group, ISBN 978-1-4398-0902-0.
- Suyanto, 2010. *Algoritma Optimasi Deterministik dan Probabilitik*, Graha Ilmu.
- Tassopoulos, X.L., dan Beligiannis, N.G., 2012, Using particle swarm optimization to solve effectively the school timetabling problem, *Soft Comput*, 1229–1252, Springer.
- Teoh, K.C., Wibowo, A. and Ngadiman, S.M., 2013. Review of state of the art for metaheuristic techniques in Academic Scheduling Problems, *Artif. Intell. OI* 10.1007/s10462-013-9399-6, Springer.