

# Sistem Informasi Manajemen Penjadwalan Kuliah Menggunakan Pendekatan Integer Programming

**Mansur**

Prodi Teknik Informatika Politeknik Negeri Bengkalis  
[mansur.polbeng82@gmail.com](mailto:mansur.polbeng82@gmail.com)

---

## Abstract

Penjadwalan kuliah merupakan masalah NP-Hard dan kompleksitas. Kompleksitas terjadi ketika jumlah mahasiswa dan pertemuan matakuliah yang harus dijadwalkan sangat banyak hingga mencapai ratusan bahkan ribuan. Tujuan penelitian adalah merancang sistem informasi penjadwalan kuliah menggunakan pendekatan *integer programming* untuk pemanfaatan resource pada perguruan tinggi politeknik. Secara umum penjadwalan kuliah dapat diselesaikan menggunakan pendekatan integer programming dengan menerapkan bilangan biner 0 dan 1. Hasil analisa data *resource* menunjukkan bahwa untuk informasi hard constraints bebas dari bentrok sedangkan *soft constraints* masih terdapat bentrok pada timeslot dosen. Hasil penelitian menggunakan integer programming dengan menggabungkan *hard constraint* dan *soft constraint* menghasilkan jadwal yang efektif namun tidak efisien dalam melakukan proses pengolahan data untuk menempatkan kelas dan dosen pada ruang dan *timeslot*, karena proses dilakukan secara manual dengan mengurutkan aktivitas yang sulit terlebih dahulu, sehingga membutuhkan waktu yang lama dalam mengatur *resource* untuk menghasilkan jadwal yang bebas dari bentrok dosen, kelas, dan ruangan. Pendekatan Integer Programming belum dapat menemukan solusi secara maksimal terutama pada pengaturan *soft constraint*.

**Keywords** : Sumberdaya Perguruan Tinggi, constraints, Penjadwalan Kuliah, *integer programming*

---

## 1. Pendahuluan

Penjadwalan merupakan suatu alokasi, berfokus pada *constraints*, dan *resource* untuk *object* yang ditempatkan pada ruang dan waktu (Pongcharoen *et al.*, 2008). Penjadwalan di perguruan tinggi merupakan masalah yang tergolong pada *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).

Untuk menghasilkan jadwal perkuliahan yang optimal pada perguruan tinggi, *constraint* sangat memiliki peranan penting dalam meningkatkan kepuasan pengguna sehingga proses pencarian yang melibatkan dosen, mahasiswa, ruang dan waktu dapat ditempatkan pada satu set timeslot yang sesuai dengan kebutuhan dan hasilnya maksimal (Kanoh dan Chen, 2013). Secara umum penjadwalan pada perguruan tinggi memiliki karakteristik yang tidak jauh berbeda dengan perguruan tinggi lainnya, sebagai contoh 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).

Secara umum penjadwalan kuliah dapat diselesaikan menggunakan pendekatan integer programming, oleh sebab itu ketika masalah tersebut muncul dan ditambah dengan *constraints* baik *hard* maupun *soft* maka perlu diselesaikan supaya dalam menghasilkan jadwal yang bebas dari bentrok dan optimal sehingga proses pelaksanaan pembelajaran dapat terlaksana dengan maksimal (Phillip *et al.*, 2014)

Tujuan dari penelitian adalah merancang sistem informasi penjadwalan kuliah menggunakan pendekatan *integer programming* untuk pemanfaatan resource pada perguruan tinggi politeknik.

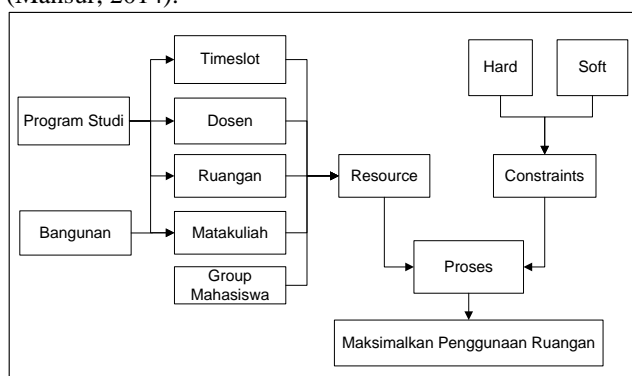
## 2. Kerangka Teori

Penjadwalan akademik pada perguruan tinggi meliputi jadwal perkuliahan dan ujian akhir. Salah satu tujuan pembuatan jadwal kuliah adalah untuk meminimalkan kesenjangan yang terjadi antara waktu dosen dan mahasiswa, sementara jadwal ujian akhir untuk memaksimalkan kesenjangan waktu dan tenaga pengawas (Teoh *et al.*, 2013).

Beberapa penelitian mengenai penjadwalan yang berfokus pada alokasi ruangan dan waktu dalam menyelesaikan masalah untuk mencapai tujuan yang telah ditentukan. Dalam penjadwalan ada dua kategori yang diperhatikan yaitu; *hard constraints* berhubungan dengan kepuasan dalam menghasilkan jadwal yang fisibel dan *soft constraints* berhubungan kualitas sistem penjadwalan. Data yang digunakan adalah data pengajar, ruangan, hari, jam mata pelajaran, jam mengajar per hari, jam mengajar perkelas, total jam mengajar dan total jam per kelas.

Semua data yang ada diasumsikan bahwa kapasitas ruangan lebih besar dari jumlah siswa (Tassopoulos dan Beligiannis, 2012).

Untuk mengoptimalkan sistem informasi penjadwalan kuliah yang efektif dan efisien maka diperlukan pengelolaan data *resource* dan *constraint* yang baik sehingga sistem informasi penjadwalan dapat menghasilkan informasi yang fleksibel sesuai dengan preferensi antara dosen, mahasiswa dan ruangan baik ruang kelas maupun laboratorium. Sistem penjadwalan kuliah dirancang dengan menggunakan metode PSO, dimana hasil penelitian menghasilkan sistem informasi manajemen penjadwalan yang dapat mengelola data *resource* tanpa menggabungkan *hard* dan *soft constraint*. Berikut komponen yang digunakan dalam membangun program penjadwalan kuliah, dapat dilihat pada Gambar 1 (Mansur, 2014).



Gambar 1. Komponen dalam membangun program penjadwalan kuliah

Penjadwalan kuliah pada universitas menggunakan *integer programming* (IP), dimana data yang digunakan seperti timeslot, ruangan, mahasiswa, dan dosen. Tujuan penelitian ini untuk meminimalkan terjadinya kesenjangan antara mahasiswa dan dosen serta menghindari terjadinya bentrok pada pembuatan jadwal. Pendekatan IP dapat digunakan untuk melakukan kustomisasi dalam mendapatkan solusi pembuatan jadwal yang optimal, namun untuk penjadwalan yang termasuk *nested partitioning* belum mungkin mendapatkan hasil yang optimal (Phillip *et al.*, 2014)

Formulasi *integer programming* untuk masalah penjadwalan disekolah, pada pendekatan *integer programming* mengusulkan penyelesaian masalah dengan menerapkan bilangan binary 0 dan 1, dimana 0 adalah *timeslot* yang masih kosong sedangkan 1 merupakan *timeslot* yang sudah dipakai dalam penempatan matakuliah beserta durasi jam yang diperlukan. Pola 0,1 dapat membantu sekolah yang berbeda dalam membuat jadwal dengan melibatkan berbagai *constraint* yang ada, selain itu pola tersebut dapat diterapkan untuk mengurangi jumlah variabel yang digunakan sehingga dapat mempermudah menyelesaikan masalah penjadwalan dengan *fleksibel* seperti menambah, menghapus, atau mengganti data sesuai kebutuhan dan menghasilkan jadwal yang efisien (Chien *et al.*, 2014).

Model *integer linear programming* untuk penjadwalan kuliah pada universitas, dimana dalam penelitian bertujuan untuk dapat membuat jadwal kuliah dengan menugaskan

dosen, ruangan, matakuliah, timeslot, dan beberapa *constraints* (*hard* dan *soft*). Dalam pencarian data untuk menemukan solusi yang layak menggunakan pencarian heuristik yang disebut *knock-out*. Hasil penelitian ini menyarankan bahwa metode heuristik lebih menjanjikan dalam proses pembuatan jadwal kuliah sebab penjadwalan merupakan masalah NP-Hard (Oladokun dan Badmus, 2008).

Penjadwalan kuliah merupakan bagian masalah *nested partitioning*. 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 (Cheu *et al.*, 2006; Staereling, 2012).

Penjadwalan pada perguruan tinggi juga merupakan masalah NP-Hard (Cheu *et al.*, 2006). Ada beberapa bagian dari proses penjadwalan yang dikatakan sebagai masalah NP-Hard (Staereling, 2012) yaitu:

a. Proses Kuliah dengan melibatkan beberapa kelas (*multiple classes*)

Pada perguruan tinggi, mahasiswa mempunyai kebebasan dalam memilih bagian perkuliahan yang mereka inginkan. Hal ini berarti penjadwalan harus dapat memperhitungkan bahwa dua perkuliahan tidak boleh diberikan pada waktu yang berbeda, dimana ada kemungkinan group mahasiswa yang ingin menghadiri kedua matakuliah tersebut. Untuk mengatasinya masalah tersebut maka kedua matakuliah harus dijadwalkan pada hari dan timeslot yang sama dengan ruangan yang berbeda. Dengan kata lain ketergantungan dari beberapa kelas dalam pembuatan jadwal membuat masalah tersebut menjadi sulit untuk dipecahkan.

b. Penggabungan Kuliah dalam satu ruangan (*merging lectures in room*)

Penggabungan kuliah dalam satu ruangan yang sama bukan merupakan suatu pilihan, namun kejadian tersebut dapat terjadi karena alasan efisiensi ruangan dan biaya. Biasanya ini terjadi pada ujian akhir semester dimana dua matakuliah yang berbeda atau sama dapat digabungkan menjadi satu ruangan dengan alasan menghemat biaya staf pengawas dan penggunaan ruang. Untuk mengatasi masalah tersebut maka secara umum jumlah kapasitas ruangan lebih besar dari jumlah mahasiswa, dimana hal ini biasanya dilakukan pada awal proses pembuatan jadwal.

c. Ketersediaan Pengajar

Pada praktiknya, setiap dosen yang tersedia pada perguruan tinggi memiliki keterbatasan dosen dalam melakukan proses pengajaran, sehingga perguruan tinggi mencari solusi dengan mendatangkan dosen tidak tetap dari perguruan tinggi lainnya. Dengan keterbatasan dosen tersebut mengakibatkan keterbatasan terhadap kesediaan waktu dosen mengajar pada hari dan *timeslot* tertentu.

Dalam praktek pelaksanaan penjadwalan, masalah penjadwalan NP-Hard solusi yang diharapkan adalah bagaimana dapat memaksimalkan jumlah kelas dan pengajar dalam penggunaan *timeslot* secara spesifik, dimana setiap pengajar membutuhkan jadwal yang akan ditempatkan pada beberapa hari tertentu sehingga setiap pengajar *timeslot* harus dapat dimaksimalkan. Dalam

memaksimalkan *timeslot* tersebut sangatlah sulit, sehingga membutuhkan pendekatan metode yang cerdas untuk dapat mengatasinya. Pada dasar, menggunakan metoda klasik dengan pendekatan *integer programming* sudah dapat diterapkan, namun tidaklah cukup untuk dapat memaksimalkan kelas dan pengajar dalam suatu *timeslot*, sehingga diperlukan algoritma heuristik yang dapat melakukan analisa secara analitis dimulai dengan mendefinisikan masalah, menentukan *constraint*, dan menentukan prosedur algoritma untuk menempatkan semua pengajar pada suatu *timeslot* yang tersedia dengan mengurutkan aktivitas yang sulit dalam pembuatan jadwal (Staereling, 2012).

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 ditugas 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.

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

Dalam menyelesaikan masalah penjadwalan ada beberapa teori dasar yang harus diperhatikan (Staereling, 2012) yaitu:

a. Masalah Kelas-Pengajar (*class-teacher problem*)

Pada dasarnya masalah tersebut merupakan adalah bagaimana menempatkan kelas dan pengajar kedalam ruang dan waktu sehingga kebutuhan proses pembelajaran menjadi layak (*hard*). Penempatan kelas dan pengajar dapat diformulasikan dibawah ini:

Diberikan satu set kelas  $C = \{c_1, \dots, c_m\}$ , satu set pengajar  $T = \{t_1, \dots, t_n\}$ , unit waktu  $P$ , dimana matrik  $R = (m \times n)$  merupakan matrik perbandingan dalam penempatan antara kelas  $c_i$  dan pengajar  $t_j$ . Sedangkan  $r_{ij}$  adalah total jumlah pengajar yang akan ditempatkan antara kelas  $c_i$  dan pengajar  $t_j$ .

Dari formulasi tersebut ada tiga *constraint* yang harus diperhatikan yaitu:

1. Setiap pelajaran membutuhkan pengajar.
2. Setiap kelas tidak boleh lebih dari satu pengajar dalam waktu yang sama.
3. Setiap dosen tidak boleh lebih dari satu memberikan pelajaran dalam waktu yang sama.

Permasalahan kelas dan pengajar lebih jelas dapat diformulasikan sebagai *integer programming* yang menggunakan variabel binary.

$$x_{ijk} = \begin{cases} 1, & \text{jika kelas } c_i \text{ dan pengajar } t_j \text{ bertemu pada satu unit waktu } k \\ 0, & \text{dan sebaliknya} \end{cases}$$

Persyaratan:

$$\sum_{k=1}^p x_{ijk} = r_{ij}, \quad i = 1, \dots, m; j = 1, \dots, n,$$

$$\sum_{j=1}^n x_{ijk} \leq r_{ij}, \quad i = 1, \dots, m; k = 1, \dots, p,$$

$$\sum_{i=1}^m x_{ijk} \leq r_{ij}, \quad j = 1, \dots, n; k = 1, \dots, p,$$

$$x_{ijk} \in \{0, 1\} \quad i = 1, \dots, m; j = 1, \dots, n; k = 1, \dots, p.$$

Dimana :

$i$  = Parameter dari kelas  $c_i$

$j$  = Parameter dari pengajar  $t_j$

$x_{ijk}$  = Total jumlah kelas dan pengajar yang bertemu pada satu unit waktu  $k$

b. Keterbatasan Ruang (*limited rooms*)

Masalah ruangan pada setiap perguruan tinggi merupakan masalah yang nyata, karena pada setiap ruangan memiliki keterbatasan *timeslot* dalam satu hari dan satu minggu. Keterbatasan ruang juga berhubungan dengan kapasitas yang tersedia, sehingga mengalami kesulitan dalam menempatkan semua kegiatan proses

pembelajaran pada ruangan yang ada. Hal tersebut dapat diformulasikan dibawah ini:

$$\sum_{i=1}^m \sum_{j=1}^n x_{ijk} \leq p$$

Dimana :

$p$  = Ruangan yang digunakan dalam penempatan kelas  $c_1$  dan pengajar  $t_j$ .

c. Maksimum Hari (*daily maximum*)

Dalam proses pembuatan jadwal, pengajar dan kelas selalu berhubungan dengan batas maksimal hari yang digunakan dalam satu minggu, idealnya memiliki 5 hari dimulai dari hari senin sampai jum'at dan satu hari terdiri dari 8 periode, sehingga dalam satu minggu berjumlah 40 *timeslot*. Setiap dosen dan kelas memiliki batas maksimum perhari dalam melakukan proses perkuliahan, sehingga mengakibatkan keterbatasan dosen. Hal tersebut dapat diformulasikan dibawah ini:

$$\sum_{j=1}^n x_{ijk} \leq a_i, \quad i = 1, \dots, m; k = 1, \dots, p,$$

$$\sum_{i=1}^m x_{ijk} \leq b_j, \quad j = 1, \dots, n; k = 1, \dots, p,$$

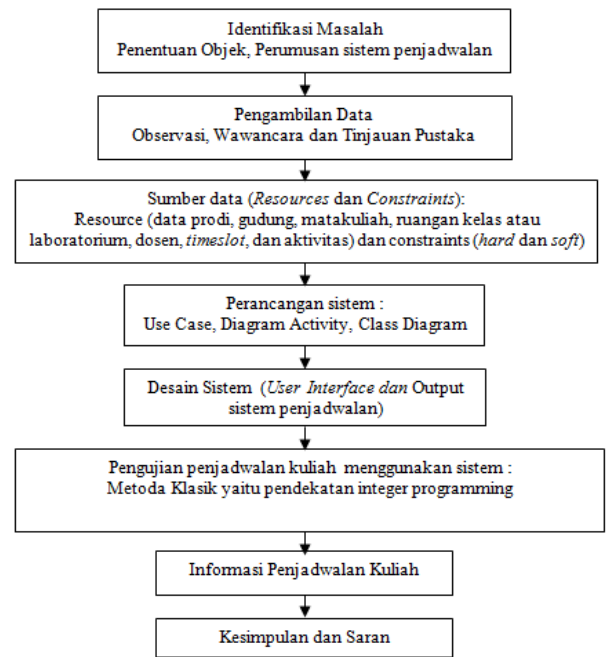
$$x_{ijk} \in N_0 \quad i = 1, \dots, m; j = 1, \dots, n; k = 1, \dots, p.$$

Dimana  $x_{ijk}$  merupakan ruang  $c_1$  dan pengajar  $t_j$  bertemu pada satu unit waktu  $k$ .  $a_i$  dan  $b_i$  adalah jumlah pengajar maksimum yang disajikan pada kelas  $c_1$  dan pengajar  $t_j$  yang mungkin terlibat pada salah satu hari  $p$ .  $N_0$  merupakan element dari  $p$ .

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 pendekatan *integer programming* sebagai pengujian sistem. Penentuan pendekatan *integer programming* merujuk pada (Staereling, 2012)

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 2014/2015 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*.



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.

Masalah penjadwalan kuliah pada prinsipnya dapat diselesaikan dengan menggunakan pendekatan *integer programming* dengan variabel binary [0, 1] yang merupakan metode klasik untuk menyusun penjadwalan kuliah secara manual. Pada kasus penjadwalan kuliah ini, satu set jadwal kuliah yang terdiri dari beberapa kelas

yang mengikuti aktivitas perkuliahan. Ada beberapa tahapan yang harus diperhatikan dalam menyusun jadwal secara klasik (Staereling, 2012) yaitu;

1. Inisialisasi data.

Menempatkan kelas  $c_1$  dan pengajar  $t_j$  pada satu unit waktu  $k$ , dimana penempatan tersebut berdasarkan persyaratan seperti setiap matakuliah membutuhkan dosen, menghindari bentrok dosen, bentrok kelas dan bentrok ruangan. Berikut formulasi pendekatan integer programming.

$$x_{ijk} = \begin{cases} 1, & \text{jika kelas } c_1 \text{ dan pengajar } t_j \text{ bertemu pada satu unit waktu } k \\ 0, & \text{dan sebaliknya} \end{cases}$$

Tabel 2. Sebaran Data Kuliah

Nama Matakuliah	Smt	Sks	Jam	ID Kls	Kelas
Teori KP 1	1	2	3	1	TI-1A
Prak.KP 1	1	2	6	2	TI-1A
Disain Web	3	2	4	3	TI-3A
Adm. Sistem	3	2	4	4	TI-3A

Tabel 3. Sebaran timeslot dan kesediaan jam mengajar dosen.

	Waktu	SN	SL	RB	KM	JT	ST
PERIODE	1 08:00 : 08:50	1	11	21	31	41	51
	2 08:50 : 09:40	2	12	22	32	42	52
	3 09:40 : 10:30	3	13	23	33	43	53
	4 10:30 : 11:20	4	14	24	34	44	54
	5 11:20 : 12:10	5	15	25	35	45	55
	6 12:10 : 13:00	6	16	26	36	46	56
	7 13:00 : 13:50	7	17	27	37	47	57
	8 13:50 : 14:40	8	18	28	38	48	58
	9 14:40 : 15:30	9	19	29	39	49	59
	10 15:30 : 16:20	10	20	30	40	50	60

Sebagai contoh data yang digunakan adalah pada Tabel 2 dan Tabel 3 dengan menyusun data aktivitas mengajar kedalam matrik. Tabel 3.5 matrik perbandingan penempatan kelas dan dosen untuk setiap dosen dengan durasi jam yang berbeda.

Tabel 4 Matrik perbandingan untuk penempatan kelas dan dosen.

Kelas\Dosen	Dosen 1	Dosen 2
Kelas TI-1A	1 (3 Jam, Lab)	1 (6 Jam, Lab)
Kelas TI-3A	1 (4 Jam, Lab)	1 (4 Jam, Lab)

2. Menyusun jadwal kuliah

Berdasarkan Tabel 3.4 maka dapat disusun jadwal kuliah dengan memperhatikan bentrok dosen, kelas, dan ruangan pada waktu yang sama. Gambar 3.3 menunjukkan bahwa penempatan kelas ke-1 pada dosen ke-1 pada hari selasa maka kelas ke-1 tidak boleh diajarkan oleh dosen ke-2, dan dosen ke-1 tidak boleh mengajar pada kelas ke-2 pada laboratorium dan timeslot yang sama sehingga dapat

menghindari bentrok dosen, kelas, ruangan, waktu istirahat sholat jum'at, dan keterbatasan dosen mengajar. Total timeslot sebanyak 60, terdiri dari enam hari dalam satu minggu dan 10 periode dalam satu hari, dimana setiap dosen memiliki kesempatan yang sama sebanyak 60 timeslot yang tersedia.

4. Hasil dan Pembahasan

4.1 Hasil Penelitian

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

Berdasarkan hasil analisa data menggunakan metoda klasik dengan mengabungkan hard constraint dan soft constraint menghasilkan jadwal yang efektif namun tidak efisien dalam melakukan proses pengolahan data untuk menempatkan kelas dan dosen pada ruang dan timeslot, karena proses dilakukan secara manual dengan mengurutkan aktivitas yang sulit terlebih dahulu, sehingga membutuhkan waktu yang lama dalam mengatur resource untuk menghasilkan jadwal yang bebas dari bentrok dosen, kelas, dan ruangan. Metoda klasik juga belum dapat bekerja secara maksimal terutama pada pengaturan soft constraint.

4.2 Pembahasan

Proses metoda klasik dalam mengelola data resource dan constraint untuk menghindari bentrok yang sudah ditentukan berdasarkan hard dan soft constraint. Input data melibatkan BAK sebagai administrator, kemudian data tersebut diproses untuk pembuatan jadwal kuliah. Pada proses klasik, terlebih dahulu mendefenisikan masalah seperti data dosen-matakuliah-kelas-ruangan-timeslot (DMKRT), menentukan hard constraint yaitu; dosen-timeslot (DT), kelas-timeslot (KT), ruangan-timeslot (RT), kapasitas ruangan, matakuliah dilaboratorium dan istirahat waktu sholat serta soft constraint yaitu; kesediaan dosen-timeslot (SDT) dan kesediaan ruangan-timeslot (SRT). Setelah selesai tahapan inisialisasi, metoda klasik mengurutkan aktivitas dosen yang paling sulit terlebih dahulu dan menempatkan aktivitas tersebut pada hari dan timeslot yang tersedia dengan mengutamakan pencarian lokal. Hasil pengolahan data dapat menghasilkan informasi jadwal kuliah. Gambar 3 merupakan langkah yang digunakan dalam melakukan penyusunan penjadwalan secara otomatis.

```

Inisialisasi :
1. Tentukan timeslot awal secara default
   (Inisialisasi Timeslot, ruang-waktu (RT),
   Kesediaan Ruangan (Default 'Y'), dan Inisialisasi
   Kesediaan Dosen (Default 'Y')).
2. Tentukan Kombinasi dari suatu partikel (DMKRT)
   - Inisialisasi setjadwal secara acak kedalam RT.
   - Hitung nilai [i].
3. Proses Pendekatan Integer Programming ():
   Procedure qinit_Jadwal
   Fori=1 sampai jumlah setjadwal
    
```



```

Reset tabel resource
Update tabel ruang-waktu
Pilih DMK yang mempunyai ruang tertentu,
urutkan berdasarkan slot waktu yang
Dibutuhkan dari besar kekecil→ tbl1
For each DMKR in tbl1
    Ambil kodeDosen-Matakuliah-Kelas-
    Ruang
    Pilih waktu yang Cocok untuk
    Matakuliah tersebut sesuai kesediaan
    waktu dosen
    pengajar→ T
    Masukkan potongan DMKRT ini ketabel
    resource
End For
Ambil DMK yang tidak memiliki aturan ruang
tertentu, urutkan berdasarkan slot
waktu yang dibutuhkan dari besar kekecil→
tbl1

For each DMK in tbl1
    Ambil kodeDosen-Matakuliah-Kelas
    Pilih Ruang dengan sisa slot waktu
    terbanyak→ R
    Pilih waktu yang cocok sesuai
    kesediaan dosen pengajar→ T
    Simpan potongan DMKRT ketabel resource
End For
Gabungkan semua potongan menjadi satu setjad
End For
For i=1 sampai jumlah setjadwal
    Hitungfitness setjadwal
End For
End Procedure
    
```

Gambar 3. Tahapan penyusunan jadwal secara otomatis menggunakan pendekatan *integer programming*

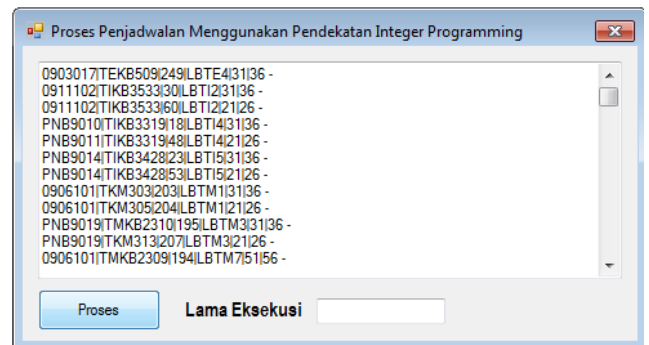
Pengujian sistem informasi penjadwalan kuliah menggunakan data *resource* dan *constraint* yang tersedia, kemudian dilanjutkan dengan proses pembuatan jadwal secara otomatis. Data rekapitulasi *resource* dapat dilihat pada Tabel 5.

Tabel 5. *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 2014-2015 (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 Bengkulu secara keseluruhan.	867 Orang
7	Jumlah gedung yang tersedia	6 Gedung
8	Jumlah ruangan kelas	20 Kelas
9	Jumlah ruangan laboratorium	34 Laboratorium

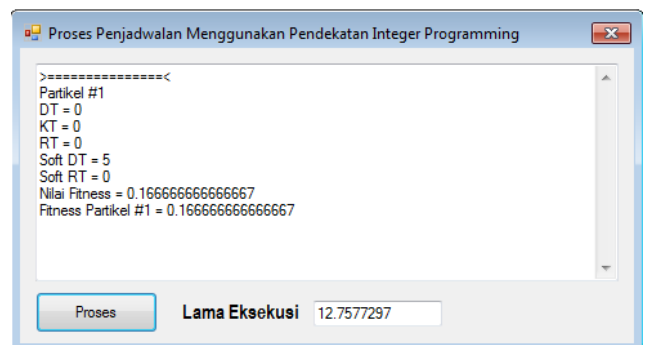
Sedangkan *constraint* yang digunakan dalam mengatur proses penjadwalan kuliah dapat dilihat pada bagian 3 (*hard* dan *soft constraint*).

Gambar 4 menunjukkan proses pengolahan data *resource* dengan menggabungkan beberapa *hard* dan *soft constraints*.



Gambar 4. Proses pengolahan data *resource*

Berikutnya Gambar 5 merupakan hasil dari proses pengolahan data *resource*, dimana gambar tersebut menampilkan informasi *hard constraints* yang bebas dari bentrok dan *soft constraints* masih terdapat bentrok pada timeslot dosen.



Gambar 5. Hasil proses pengolahan data *resource*

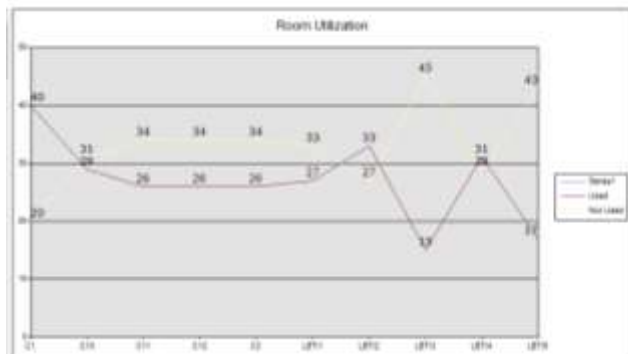
Gambar 6 menunjukkan hasil proses pengolahan data *resource* menggunakan pendekatan *integer programming* yang menghasilkan informasi jadwal kuliah.

No	Jam	TEKNIK INFORMATIKA						
		Dosen	Mata Kuliah	Dosen/Instruktur	Tempat	Tipe	Mata Kuliah	
Dosen	08:00 - 09:00	IA	Prologika 1	Has Sunardi, S.Pd	0.2	08	Tanggal, Logika	
	09:00 - 10:00	IA	Prologika 1	Has Sunardi, S.Pd	0.2	08	Tanggal, Logika	
	10:00 - 11:00	IA	0.T1	Linda Permatasari, S.Pd	Jab.Peng.Mul	08	Tanggal, Logika	
	11:00 - 12:00	IA	0.T1	Linda Permatasari, S.Pd	Jab.Peng.Mul	08	Prologika 1	
	12:00 - 13:00	IA	0.T1	Linda Permatasari, S.Pd	Jab.Peng.Mul	08	Prologika 1	
	13:00 - 14:00	IA	0.P.2	Das Ardiyanah, S.Ners	Jab.Peng.Mul	08	Manajemen	
	14:00 - 15:00	IA	0.P.2	Das Ardiyanah, S.Ners	Jab.Peng.Mul	08	Manajemen	
	15:00 - 16:00	IA	0.P.2	Das Ardiyanah, S.Ners	Jab.Peng.Mul	08	Manajemen	
	Mata Kuliah	08:00 - 09:00	IA	0.P.2	Das Ardiyanah, S.Ners	Jab.Peng.Mul	08	Manajemen
		09:00 - 10:00	IA	0.P.2	Das Ardiyanah, S.Ners	Jab.Peng.Mul	08	Manajemen
10:00 - 11:00		IA	0.P.2	Das Ardiyanah, S.Ners	Jab.Peng.Mul	08	Manajemen	
11:00 - 12:00		IA	0.P.2	Das Ardiyanah, S.Ners	Jab.Peng.Mul	08	Manajemen	
12:00 - 13:00		IA	0.P.2	Das Ardiyanah, S.Ners	Jab.Peng.Mul	08	Manajemen	
13:00 - 14:00		IA	0.P.2	Das Ardiyanah, S.Ners	Jab.Peng.Mul	08	Manajemen	
14:00 - 15:00		IA	0.P.2	Das Ardiyanah, S.Ners	Jab.Peng.Mul	08	Manajemen	
15:00 - 16:00		IA	0.P.2	Das Ardiyanah, S.Ners	Jab.Peng.Mul	08	Manajemen	

Gambar 6. Jadwal kuliah menggunakan pendekatan *integer programming*

Gambar 7 menunjukkan penggunaan ruangan baik ruang kelas atau laboratorium, dimana gambar tersebut dapat memberikan informasi kepada dosen, program studi

dan pimpinan mengenai jumlah jam pemakaian ruangan dalam satu minggu. Informasi tersebut juga dapat digunakan sebagai bahan pertimbangan pimpinan dan program studi dalam merencanakan pengembangan prodi menjadi lebih baik.



Gambar 7. Pemanfaatan ruangan.

## 5. Kesimpulan

Penjadwalan secara umum dapat diselesaikan menggunakan pendekatan integer programming dengan menerapkan bilangan binary 0 dan 1, dimana 0 adalah *timeslot* yang masih kosong sedangkan 1 merupakan *timeslot* yang sudah dipakai dalam penempatan matakuliah beserta durasi jam yang diperlukan. Dalam proses implementasi masalah penjadwalan merupakan masalah NP-Hard sehingga sulit untuk menemukan solusi yang terbaik, karena semakin jumlah dosen, kelas banyak maka tingkat kompleksitas semakin tinggi.

Sistem informasi 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 pendekatan *integer programming* untuk membantu mengoptimalkan penggunaan ruangan dalam proses pembuatan jadwal kuliah secara otomatis sehingga proses pelaksanaan pembelajaran dapat terlaksana dengan maksimal.

Hasil analisa data menggunakan pendekatan *integer programming* dengan mengabungkan *hard constraint* dan *soft constraint* menghasilkan jadwal yang efektif namun tidak efisien dalam melakukan proses pengolahan data untuk menempatkan kelas dan dosen pada ruang dan

*timeslot*, karena proses dilakukan secara manual dengan mengurutkan aktivitas yang sulit terlebih dahulu, sehingga membutuhkan waktu yang lama dalam mengatur *resource* yang tersedia. Dari penelitian ini, masih perlu pengembangan terutama dalam hal penambahan metode heuristik yang lain serta menggabungkan antara hard dan soft constraint sehingga preferensi mahasiswa dalam menentukan pembuatan jadwal kuliah dapat terlaksana dengan optimal.

## Daftar Pustaka

- Chien, N.N, Thanh, X.L, dan Truong, L.H, 2014, An integer programming formulation for a class of real-life school timetabling problems, *European Journal of Operational Research*, Hanoi, Vietnam.
- Chu, C.S., Chen, T.Y., dan 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.
- Irene, F.S.H., Deris, S., dan Hashim, M.Z.S., 2009, Incorporating Of Constraint-Based Reasoning Into Particle Swarm Optimization For University Timetabling Problem, *ISSR Journals*, Vol. 1 (1).
- Kanoh, H., dan Chen, S., 2013, Particle Swarm Optimization with Transition Probability for Timetabling Problems, *LNCS*, Springer.
- Mansur, 2014, Perancangan Sistem Informasi Penjadwalan Resource Perguruan Tinggi Menggunakan Metode Particle Swarm Optimization (PSO), *Inovasi dan Teknologi Politeknik Negeri Bengkalis (Inovtek Polbeng)*, Volume 4, Nomor 2, Oktober, hlm. 75-86.
- Oladokun, V.O dan Badmus, S.O, 2008, An Integer Linear Programming Model of a University Course Timetabling Problem, *The Pacific Journal of Science and Technology*, Volume 9. Number 2.
- Phillips, E.A, Walker, G.C, Ehr Gott, M, dan Ryan, M.D, 2014, Integer Programming for Minimal Perturbation Problems in University Course Timetabling, *International Conference of the Practice and Theory of Automated Timetabling*, PATAT 2014, 26-29, United Kingdom.
- Pongcharoen, P., Promtet, W., Yenradee, P., dan Hicks, C., 2008, Stochastic Optimisation Timetabling Tool for university course scheduling, *Int. J. Production Economics* 112 903-918, Elsevier.
- 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.
- 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., dan 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.