

Implementasi Penjadwalan Kuliah *Job Shop* Dengan Perancangan Jadwal Kuliah Menggunakan *Constraints Programming*

Firdaus

STMIK Amik Riau

Jl. Purwodadi Indah KM.10, Sidomulyo Barat, Tampan, Pekanbaru

firdauskoto@stmik-amik-riau.ac.id

Abstrak — Salah satu permasalahan utama pada bagian administrasi prodi/jurusan STMIK Amik Riau adalah permasalahan membangun jadwal kuliah. Inti dari permasalahan ini adalah menjadwalkan sejumlah komponen yang terdiri atas mahasiswa, dosen, ruang dan waktu dengan sejumlah batasan atau syarat tertentu. Penelitian ini bertujuan untuk menghasilkan suatu jadwal kuliah menggunakan metode *constraints programming* yang memenuhi *hard constraints*. Langkah awal yang dilakukan adalah mempresentasikan masalah penjadwalan sebagai sebuah *constraints satisfaction problem* (CSP), kemudian proses pencarian solusi CSP menggunakan *blind search algorithm* yaitu menguji seluruh ruang solusi yang ada. Hasil penelitian adalah suatu slot jadwal kuliah yang memenuhi *constraints* sehingga tidak terjadi bentrok dalam menyusun jadwal perkuliahan.

Kata Kunci — penjadwalan kuliah, *constraint programming*, *constraint satisfaction problem*

I. PENDAHULUAN

Permasalahan penjadwalan *job shop* bukanlah suatu hal yang asing. Banyak hal yang berhubungan dengan masalah penjadwalan salah satunya masalah penjadwalan kuliah. Penjadwalan kuliah harus disusun secara baik dan teliti dengan meminimalkan kemungkinan palanggaran, karena hasil pembuatan jadwal sangat berpengaruh pada kualitas kegiatan perkuliahan. Masalah penjadwalan kuliah ini telah menarik minat banyak orang untuk membuat aplikasi dengan berbagai metode dan algoritma agar mendapatkan hasil yang optimal.

Rumitnya menyusun penjadwalan kuliah merupakan masalah yang sering terjadi di perguruan tinggi, dimana setiap memasuki semester baru mahasiswa mendapatkan jadwal yang baru. Banyak hal yang harus dipertimbangkan dalam penyusunan jadwal kuliah baru terutama sulitnya mengatur waktu antara mahasiswa, dosen, mata kuliah, ruang dan waktu. Proses penjadwalan secara manual sangat rumit dan besar kemungkinan adanya pelanggaran *constraints* yaitu pelanggaran pada batasan yang telah ditentukan dalam penyusunan jadwal kuliah yang berakibat kurang optimalnya mutu pendidikan akademik pada perguruan tinggi. Akibat terjadinya *constraints* maka proses penjadwalan kuliah perlu di susun kembali, hal ini berdampak buruk untuk kelancaran perkuliahan di *fase* awal kuliah.

Penyusunan jadwal perkuliahan pada STMIK Amik Riau masih menggunakan cara manual. Hal ini kurang efektif karena jadwal kuliah dikerjakan secara manual dan banyak kekurangan yang berakibat timbulnya pelanggaran *constraint* oleh kesalahan manusia. Sehingga perlu dilakukan proses penyelesaian penjadwalan dengan

menggunakan metode *constraints programming*. Menurut Talbi yang melakukan penelitian penjadwalan menggunakan metode *heuristic* dimana metode tersebut adalah seni untuk menemukan strategi dalam menyelesaikan persoalan [23].

Saat ini telah berkembang berbagai macam teknik *heuristic search* untuk menyelesaikan masalah *optimasi* terutama masalah penjadwalan *job shop* salah satunya yaitu *constraint satisfaction problem* yang menyelesaikan model permasalahan dengan melihat berbagai *constraints* yang akan dihadapi, oleh karena itu penelitian ini dilakukan untuk melihat bagaimana hasil penyelesaian permasalahan pada penjadwalan kuliah dengan metode *constraints programming*.

Berdasarkan uraian dari latar belakang masalah tersebut, maka permasalahan nya antara lain adalah :

1. Bagaimana merancang model penjadwalan dalam masalah jadwal kuliah dengan metode *constraint programming* ?
2. Bagaimana model penjadwalan dirancang untuk kebutuhan manajemen dan sistem terintegrasi dengan informasi ?

Berdasarkan uraian masalah di atas diharapkan pemodelan berorientasi objek dalam penjadwalan kuliah ini mampu memberikan gambaran terhadap keuntungan penggunaan sistem yang telah dirancang dengan harapan peralihan ke teknologi dari konvensional ke model dapat dilaksanakan.

II. TINJAUAN PUSTAKA

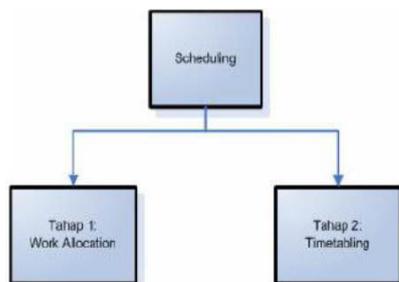
A. Penjadwalan *Job-Shop*

Penjadwalan merupakan suatu proses untuk membuat sebuah jadwal dimana seseorang berada dalam satu tempat

dengan waktu yang telah ditetapkan. Penjadwalan merupakan suatu cara yang dilakukan oleh manusia agar bisa memanfaatkan waktu dengan efisien sehingga dapat mencapai suatu tujuan rutinitas yang maksimal. Proses pembuatan jadwal harus dilakukan dengan ketelitian yang tinggi dari orang yang akan membuat jadwal tersebut karena batasan (*constraints*) yang perlu diperhatikan untuk menghindari beradunya jadwal [8].

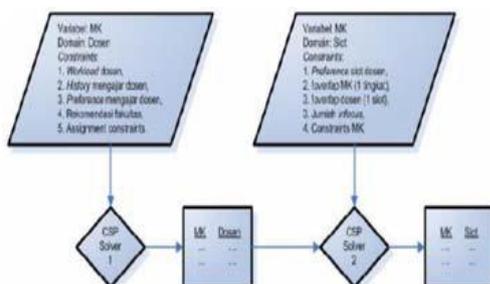
Penjadwalan juga didefinisikan sebagai suatu permasalahan dalam penentuan jadwal yang tepat atas suatu pekerjaan terhadap sumber daya yang tersedia sesuai dengan *constraints* yang harus dipenuhi [19].

Pada penelitiannya, soraya membagi kedalam 2 tahap untuk penyelesaian penjadwalan kuliah. Tahap pertama *work allocation*, yaitu pemetaan mata kuliah dan dosen yang memenuhi *constraints*. Tahap kedua *timetabling*, yaitu pemetaan mata kuliah ke slot (hari dan jam kuliah) yang memenuhi *constraints*. Berikut tahapan model penjadwalan kuliah dapat dilihat pada gambar 1.



Gambar 1. Tahapan Model Penjadwalan Kuliah

Proses yang dilakukan pada tahap awal adalah pemetaan matakuliah yang akan diajar terhadap dosen yang mengajar. Permasalahn tahap 1 ini dimodelkan ke dalam bentuk *constraint satisfaction problem* (CSP). Proses yang dilakukan pada tahap kedua yaitu pemetaan mata kuliah yang akan diajar kepada slot perkuliahan. Permasalahan tahap 2 ini juga dimodelkan ke dalam bentuk *constraints satisfaction problem* (CSP). Soraya menggunakan teknik *branch and bound* dalam mencari solusi dari model CSP pada tahap 1 dan tahap 2. Definisi variable, *domain*, dan *constraints* tahap 1 dan 2 dapat dilihat pada gambar 2.



Gambar 2. Alur model penjadwalan kuliah

Agar dapat menyelesaikan masalah penjadwalan kuliah yang dimodelkan ke dalam bentuk *constraints satisfaction problem* (CSP) maka digunakan metode *constraints programming*. penjelasan metode *constraints programming* dapat dilihat pada sub bab B. [13]

B. Konvensional dan Heuristik

1) Metode Konvensional

Metode konvensional berupa algoritma yang menggunakan perhitungan matematis biasa. Ada beberapa metode konvensional yang biasa digunakan untuk melakukan pencarian jalur terpendek, diantaranya algoritma Dijkstra, algoritma Floyd-Warshall, dan algoritma Bellman-Ford [12].

2) Metode Heuristik

Metode *heuristik* adalah sub bidang dari kecerdasan buatan yang digunakan untuk melakukan pencarian dan penentuan jalur terpendek. Ada beberapa algoritma pada metode *heuristik* yang biasa digunakan dalam pencarian jalur terpendek diantaranya yaitu algoritma semut dan algoritma genetika [12].

C. Konsep Dasar Constraint Programming

1) Constraint

Constraints merupakan beberapa macam batasan yang ditentukan dan menjadi kendala yang mana harus dikurangi pelanggaran terhadap batasan tersebut. Dalam masalah penjadwalan khususnya penjadwalan kuliah memiliki *hard constraint* dan *soft constraint*. *Hard constraints* merupakan suatu kondisi yang utama harus dipenuhi dan apabila dilanggar maka hasilnya tidak *valid*. *Soft constraints* merupakan suatu kondisi atau syarat yang masih bisa dilanggar, namun diharapkan kondisi tersebut terpenuhi untuk hasil penjadwalan yang optimal.

2) Constraint Programming

Constraints programming adalah pembelajaran sistem komputasi berdasarkan *constraints*. Ide utama dari *constraints programming* ini adalah menyelesaikan masalah dengan menyatakan *constraints* (kondisi, sifat, kebutuhan) yang harus dipenuhi oleh solusi. Dengan kata lain, *constraints programming* adalah pendekatan alternatif himpunan kebutuhan (*constraints*) yang berurutan diselesaikan metode umum ataupun spesifik untuk domain [3].

Pada *constraint programming* itu sendiri terdapat tiga tahapan penting yang harus selalu ada yaitu :

1. Proses mendeklarasikan domain dari setiap variable yang telah ditentukan.
2. Mendeskripsikan atau memodelkan *constraints problem*.
3. Mencari solusi yang tepat dengan menggunakan *backtrack search*, atau mencari solusi yang optimal berdasarkan *branch and bound search* [11].

D. Constraint Satisfaction Problem

Constraint Satisfaction Problem (CSP) secara formal dapat didefinisikan sebagai permasalahan yang direpresentasikan dalam bentuk himpunan variable dan syarat-syarat (*constraint*) [17].

Pada *Constraint Satisfaction Problem* (CSP) terdapat variable yang memiliki *domain* yang berisikan masing-masing nilai. Domain itu sendiri pada CSP sering memiliki anggota bilangan bulat. Namun ada juga yang domainnya berupa simbol.

E. Definisi dan Tujuan Penjadwalan

Penjadwalan juga didefinisikan sebagai proses pengalokasian sumber-sumber atau mesin-mesin yang ada untuk menjalankan sekumpulan tugas dalam jangka waktu tertentu.

Dalam proses pengambilan keputusan yang memberikan pengertian dalam fungsi Penjadwalan (nilai konseptual). Tujuan penjadwalan umumnya adalah sebagai berikut:

1. Meningkatkan produktifitas mesin, yaitu dengan mengurangi waktu mesin menganggur.
2. Mengurangi persediaan barang setengah jadi dengan jalan mengurangi jumlah rata-rata pekerjaan yang menunggu dalam antrian suatu mesin karena mesin tersebut sibuk.
3. Mengurangi keterlambatan suatu pekerjaan. Setiap pekerjaan mempunyai batas waktu (*due date*) penyelesaian, jika pekerjaan tersebut diselesaikan melewati batas waktu yang ditentukan maka pekerjaan tersebut dinyatakan terlambat. Dengan metoda penjadwalan maka keterlambatan ini dapat dikurangi, baik waktu maupun frekuensi.

F. Konsep Dasar Penjadwalan Perkuliahan

Secara umum penjadwalan perkuliahan adalah pengalokasian waktu dan tempat untuk suatu kegiatan perkuliahan yang pada umumnya memiliki batasan-batasan yang harus dipenuhi. Secara spesifik, masalah penjadwalan perkuliahan untuk mata pelajaran dan ujian akhir dapat dilihat sebagai perbaikan urutan waktu dan ruangan pertemuan antara pengajar dengan mahasiswa, sambil secara sekaligus memenuhi jumlah dari variasi kondisi atau batasan pokok dan batasan preferensial (Wren, 1996).

G. Searching Algorithm

Algoritma pencarian (*searching algorithm*) adalah algoritma yang menerima sebuah argument kunci dengan langkah-langkah tertentu dalam mencari rekaman dengan kunci tersebut. Setelah proses pencarian dilaksanakan, akan diperoleh salah satu dari dua kemungkinan, yaitu data yang dicari ditemukan (*successful*) atau tidak ditemukan (*unsuccessful*) [14]. Ada dua macam teknik pencarian yaitu

pencarian sekuensial dan pencarian biner. Perbedaan dari dua teknik ini terletak dari pada keadaan atau bentuk data. Pencarian sekuensial digunakan apabila data dalam keadaan acak atau tidak berurut. Sebaliknya, pencarian biner digunakan pada data yang sudah berurutan.

E. Blind Search

Blind search atau disebut juga pencarian buta merupakan sekumpulan prosedur atau cara yang digunakan untuk melacak ruang solusi yang mungkin. Proses pencarian berlangsung sampai solusi tersebut ditemukan. Adapun tujuannya yaitu menguji seluruh kemungkinan solusi yang ada untuk menemukan solusi yang memenuhi. Pencarian ini disebut pencarian buta dikarenakan memang tidak ada informasi awal yang digunakan pada saat proses pencariannya [22].

Meskipun teknik pencarian *blind search* tidak praktis untuk permasalahan yang besar namun memberikan landasan untuk memahami strategi pencarian informasi [2]. Pencarian *Blind Search* terdiri atas pencarian melebar pertama (*Breadth First Search*) dan pencarian mendalam pertama (*Depth First Search*) [9].

III. PEMBAHASAN

A. Analisa Masalah

STMIK Amik Riau terdiri dari 2 prodi/jurusan. Masing-masing prodi/jurusan memiliki slot perkuliahan yang padat. Hal ini dilihat dari jumlah mata kuliah yang tidak sedikit. Sehingga apabila jadwal perkuliahan di buat secara manual, jadwal yang tersusun mungkin belum optimal (banyak jadwal yang beradu).

Berikut ini merupakan data jumlah mata kuliah, dosen, ruang yang tersedia di STMIK Amik Riau semester ganjil tahun ajaran 2015/2016 :

Tabel 1. Data jumlah mata kuliah, dosen, ruang di STMIK Amik Riau

No	Jurusan	Jumlah Mata Kuliah				Total Jumlah MK	Total Jumlah Dosen	Total Jumlah Ruang
		Semester 1	Semester 3	Semester 5	Semester 7			
1	Teknik Informatika	9	9	8	8	34	50	20
2	Manajemen Informatika	10	10	8	-	28		

Tabel 1. adalah jumlah mata kuliah, jumlah dosen dan jumlah ruang pada STMIK Amik Riau, namun jumlah data yang diinputkan dalam penelitian ini yaitu sebanyak 114 mata kuliah, 50 dosen, 20 ruang dan 20 slot/waktu

Adapun *hard constraints* yang digunakan dalam penelitian ini yaitu:

1. Dosen tidak boleh berada pada ruang yang berbeda pada saat atau waktu yang sama.

2. Pada ruangan yang sama dan slot yang sama tidak boleh ada dua mata kuliah yang berbeda.

B. Perancangan Sistem Penjadwalan Kuliah

1) Definisi Variabel, Constraint dan Domain Penjadwalan Kuliah sebagai CSP (Constraints Satisfaction Problem)

a. Variabel

Variabel CSP yang digunakan pada penelitian ini terdiri dari domain (mk_i, d_i, r_i, s_i) dimana mki merupakan mata kuliah (MK), di adalah dosen, ri yang dinyatakan sebagai ruang, dan si sebagai slot/waktu.

b. Constraints

Tahapan awal dalam menetapkan constraints untuk penjadwalan kuliah pada penelitian ini telah dibahas sebelumnya pada bab 3. Berikut ini definisi variable secara matematis dan penjelasan masing-masing constraints merujuk pada penelitian [24]

$M = \{ \text{mata kuliah}_1, \text{mata kuliah}_2, \dots, \text{mata kuliah}_p \}$

$D = \{ \text{dosen}_1, \text{dosen}_2, \dots, \text{dosen}_q \}$

q = jumlah dosen

$R = \{ \text{ruang}_1, \text{ruang}_2, \dots, \text{ruang}_t \}$

t = jumlah ruang

$S = \{ \text{slot}_1, \text{slot}_2, \dots, \text{slot}_u \}$

u = jumlah slot

variable CSP : X_1, X_2, \dots, X_n

n = total jumlah MK yang ditawarkan pada suatu semester

l = bilangan bulat

$1 \leq i, j \leq n; i, j \in l$

$X_i = \{ (mk_i, d_i, r_i, s_i) \mid mk_i \in M, d_i \in D, r_i \in R, s_i \in S \}$

V_i = untuk setiap bilangan asli

i, j = bilangan asli

mk_i = mata kuliah

d_i, d_j = dosen

r_i, r_j = ruang

s_i, s_j = slot/waktu

Hard Constraints (HC)

HC1. Dosen tidak boleh berada pada ruang yang berbeda pada saat atau waktu yang sama

Hard Constraints 1 secara matematis yaitu :

$V_i, d_i \neq j \wedge (d_i \neq d_j \vee s_i \neq s_j)$

Seorang dosen tidak boleh mengajar mata kuliah yang berbeda pada waktu yang sama. Baik ditempat yang berbeda atau mata kuliah yang berbeda.

HC2. Pada ruangan yang sama dan slot yang sama tidak boleh ada dua mata kuliah yang berbeda.

Hard Constraints 2 secara matematis yaitu :

$V_i, j_i \neq j \wedge (r_i \neq r_j \vee s_i \neq s_j)$

Pada ruangan dan slot yang sama, tidak boleh ada mata kuliah yang berbeda karena akan beradu jadwal. Misalnya Ruang 1 pada slot pukul 08:00 hanya boleh terjadwal satu mata kuliah. Jika tidak, maka jadwal akan beradu dan ruang tidak dapat digunakan.

c. Domain

Indeks nilai dari masing-masing domain untuk mki (mata kuliah), di (dosen), ri (ruang), si (slot) yang dinyatakan dalam bentuk kode angka i € bilangan asli. Contoh kode nilai tersebut dapat dilihat pada tabel 2. di bawah :

Tabel 2. Contoh informasi data mata kuliah STMIK Amik Riau

Kode/Id	Mata Kuliah	Semester	Jurusan
1101	Algoritma & Pemrograman	1	Managemen Informatika
1307	Sistem Operasi	3	Managemen Informatika
1403	Etika & Profesi	4	Managemen Informatika
1601	Multimedia	6	Managemen Informatika
1201	Kalkulus 1	1	Teknik Informatika
1204	Basis Data 1	2	Teknik Informatika
1207	Struktur data	2	Teknik Informatika
1302	Aljabar Linier	3	Teknik Informatika
1406	Statistik Probabilitas	4	Teknik Informatika
1402	Teknik Digital	4	Teknik Informatikaa
1526	Metode Penelitian	5	Teknik Informatika
1820	Perbankan	8	Teknik Informatika

Tabel 2. adalah contoh informasi data mata kuliah STMIK Amik Riau yang di input ke database. Pada tabel tersebut terdapat informasi mata kuliah dengan kode/id masing - masing. Berikut setiap informasi pada mata kuliah seperti table 2 di atas yaitu:

- Mata kuliah
- Kode/ id mata kuliah
- Semester
- Jurusan

Table 3. Contoh informasi data dosen STMIK Amik Riau

Kode/Id	Nama	Jurusan
1	Firdaus, M.Kom	Teknik Informatika
5	Dr Erlin, M.Kom	Teknik Informatika
10	Unang Rio, M.Kom	Teknik Informatika
19	Turkis Nasution, M.Kom	Teknik Informatika
20	Edwar Ali, M.Kom	Teknik Informatika
25	Rahmaddeni, M.Kom	Teknik Informatika
39	Helda Yenni, M.Kom	Manajemen Informatika
42	Abrar Hadi, M.Kom	Manajemen Informatika

Tabel 3. adalah contoh informasi data dosen Teknik Informatika. Setiap dosen memilik informasi yaitu :

- Kode/ id dosen
- Nama dosen
- Jurusan

Table 4. Contoh informasi data ruangan

Kode/Id	Nama
1	R1 (Ruang 1)
2	R2 (Ruang 2)
3	R3 (Ruang 3)
4	R4 (Ruang 4)
5	R5 (Ruang 5)
6	R6 (Ruang 6)
7	R7 (Ruang 7)

Tabel 4.4 adalah contoh informasi data ruangan STMIK Amik Riau. Setiap ruangan memiliki informasi yaitu :

- Kode/ id ruang
- Nama ruang

Table 5 Contoh informasi slot/waktu

Kode/Id	Jam	Hari
1	08:00	Senin
2	10:00	Senin
3	13:00	Senin
4	17:00	Selasa
5	08:00	Rabu
6	13:00	Kamis
7	10:00	Jumat

Tabel 5. adalah contoh informasi slot/waktu. Setiap slot/waktu memiliki informasi yaitu :

- Kode/id slot/waktu
- Nama hari slot/waktu perkuliahan berlangsung
- Jam kapan slot/waktu perkuliahan berlangsung

Table 6. informasi mk_dosen

id_mk	Id_dosen
1	1
11	1
15	2
16	2
3	3

Tabel 6. adalah contoh informasi data mk_dosen STMIK Amik Riau. Setiap mk_dosen memiliki informasi yaitu :

- id_mk yaitu penunjukan kode mata kuliah
- id_dosen yaitu penunjukan kode dosen

Berdasarkan table 2, table 3, table 4, table 5, table 6 dapat dilihat bahwa contoh informasi data mata kuliah STMIK Amik Riau, informasi data dosen, informasi data ruangan, informasi slot waktu dan informasi mk_dosen saling berhubungan dan memiliki kode/id masing-masing.

Pada kasus STMIK Amik Riau, penjadwalan kuliah dapat dimodelkan sebagai *constraints Satisfaction Problem*

(CSP) dengan variable X_1, X_2, \dots, X_n dimana n yaitu total jumlah mata kuliah yang ditawarkan pada suatu semester dan $X_i = \{mki, di, ri, si \mid 1 \leq i \leq n, i \in \text{bilangan bulat}, mki \in \text{mata kuliah}, di \in \text{dosen}, \in \text{ruang}, si \in \text{slot/waktu}\}$ dengan pengertian yaitu mata kuliah mki diajar oleh dosen di diruang ri pada slot/waktu si [24].

Sebagai contoh misalnya (1,1,1,1,) artinya adalah mata kuliah kalkulus I (A) diajar oleh dosen Drs. Suprasman, di ruang R1 pada jam 08:00. Pada penelitian ini dapat dilihat pada table 4.5 bahwa setiap id_mk dihubungkan dengan id_dosen karena dosen ditetapkan mengajar sesuai dengan bidang ilmunya. Sebagai contoh yaitu id_mk (1) dengan id_dosen(1) artinya MK dengan kode/id 1 diajar oleh dosen Drs. Suprasman dengan kode/id 1.

2) Pemrograman Penjadwalan Kuliah Menggunakan *Constraints Programming*

Setelah di modelkan masalah penjadwalan ke dalam *constraints satisfaction problem*, kemudian dibuat program untuk penjadwalan kuliah menggunakan metode *constraints programming*.

1. Input

Input dari program adalah file-file yang berisi informasi tentang mata kuliah, dosen, ruang, waktu. Masing-masing informasi disimpan dalam satu database. Selanjutnya dibahas mengenai masing-masing data yang di input, keteranganya, dan contoh data yang bersangkutan dapat dilihat pada gambar 3 contoh input data mata kuliah, artinya mata kuliah dengan nama kalkulus I (A), semester dua, jurusan Teknik Informatika memiliki id KB-1101.

Gambar 3. Contoh input data mata kuliah

Informasi data dosen yang disimpan ke dalam database dapat dilihat pada gambar 4 Contoh data dosen artinya dosen dengan nama Firdaus, M.Kom dosen jurusan Teknik Informatika memiliki id 1

No	NIKD	Nama Lengkap	Email	Handphone	Prodi
1	1020509102	FIRDAUS, S KOM, M KOM	firdausfa@gmail.com	082171882820	TEKNIK INFORMATIKA
2	10130509101	RADYAN RAHMA, S KOM, M KOM			TEKNIK INFORMATIKA
3		URANG RO, S KOM, M KOM			TEKNIK INFORMATIKA
4		ANTON FIRMANTO, S KOM, M KOM			TEKNIK INFORMATIKA
5		JOWAR ARIF, MMI			TEKNIK INFORMATIKA
6		DR. ERLAN, M KOM			TEKNIK INFORMATIKA
7		TURANS NASUTION, S KOM, M KOM			TEKNIK INFORMATIKA
8		EDWAR ALI, S KOM, M KOM			TEKNIK INFORMATIKA
9	1012048001	RAHMADDEEN, S KOM, M KOM	rahmaddeem@stmik-mkn-mkn.ac.id	08121542323	TEKNIK INFORMATIKA

Gambar 4. Contoh dosen

mk_id	Dosen_id
1	1
11	1
15	2
16	2
3	3
42	3

Gambar 7. Contoh mk_dosen

Informasi data ruang yang disimpan kedalam database dapat dilihat pada gambar 5. contoh ruang artinya ruang dengan nama ruang R1 memiliki id 5 dan terletak pada gedung 1.

No	Nama Ruangan	Kapasitas	Nama Gedung	Keterangan
1	LBR APLIKASI	30 Orang	GEDUNG 2	LAB APLIKASI
2	LBR JARINGAN	30 Orang	GEDUNG 2	LAB JARINGAN
3	LBR MULTIMEDIA	30 Orang	GEDUNG 2	LAB MULTIMEDIA
4	LBR PROGRAMING	30 Orang	GEDUNG 2	LAB PROGRAM
5	R1	30 Orang	GEDUNG 1	
6	R10	30 Orang	GEDUNG 1	
7	R11	30 Orang	GEDUNG 1	
8	R12	30 Orang	GEDUNG 1	
9	R13	30 Orang	GEDUNG 1	

Gambar 5. Contoh ruangan

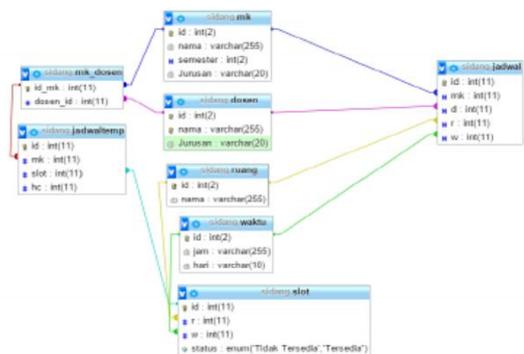
Informasi data slot/waktu yang disimpan ke dalam database dapat dilihat pada gambar 6 arti dari data pada gambar 6 contoh slot/waktu, mirip dengan arti pada table 6 yaitu kode slot, nama jam, nama hari yang artinya pada jam 08:00 dihari senin dengan kode 1

waktu_id	jam_mulai	jam_selesai	hari
1	08:00	09:45	SENEN
2	10:00	11:45	SENEN
3	13:00	15:00	SENEN
4	17:00	19:00	SELASA
5	19:30	22:00	RABU

Gambar 6. Contoh slot/waktu

Informasi data mk_dosen yang disimpan ke dalam database dapat dilihat pada gambar 7 contoh mk_dosen artinya id_mk dengan id 1 diajar oleh dosen yang memiliki id 1.

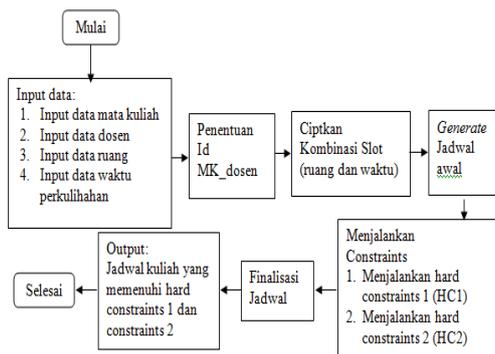
2. Proses Penyelesaian CSP (*Constraint Satisfaction Problem*)
Design struktur table penjadwalan



Gambar 8. Contoh dosen

Garis-garis penghubung antar table menunjukkan hubungan/relasi antar variable. Table mk, dosen, ruang, dan waktu mempunyai field id sebagai *primary key*. *Primary key* yang akan disimpan ke dalam table lain yang mempunyai hubungan dengan table itu masing-masing. Misalnya table jadwal dan tabel batasan. Data yang disimpan dalam dua tabel tersebut adalah id (*primary key*) dari masing-masing entitas/tabel. Garis-garis itu menunjukkam hubungan antara tabel utama dan tabel yang menjadi hubungan bagi setiap tabel utama.

Diagram alir penyelesaian masalah penjadwalan dan menemukan solusi dapat dilihat pada gambar 9. yaitu :



Gambar 9. Diagram alir proses penyelesaian *Constraint Satisfaction Problem (CSP)*

3) Hasil Pengujian Program dengan *Constraint Programming*

Berdasarkan program yang telah dibuat didapatkan output berupa jadwal kuliah yang memenuhi *constraints* yang telah ditetapkan pada penelitian ini. Hasilnya dapat dilihat pada gambar 10 dibawah ini :

No	Hari	Kode	Deskripsi	SSK	Ruang	Jam	Dosen
1	SENIN	KK-1108	FISIKAS DASAR II / PHYSICS I	2	B1	08.00 - 09.45	FRS
2	SENIN	KK-1109	PENDIDIKAN PROFESIONAL / NATIONALISM	3	B1	10.00 - 12.30	DR
3	SENIN	KK-1109	KALKULUS II / CALCULUS I	2	B2	11.00 - 14.00	ANT
4	SENIN	KK-1110	ALGORITMA & PROGRAMING / ALGORITHM & PROGRAMING	3	B12	17.00 - 19.30	JOP
5	RABU	KK-1114	PERANGKAT LEMAH APLIKASI / APPLICATION SOFTWARE I	2	B8	19.30 - 20.15	DR
6	SELASA	KB-1109	PENSANTAR TEKNOLOGI INFORMASI / INFORMATION TECHNOLOGY	2	B11	08.00 - 09.45	DRP
7	RABU	KK-1124	BANJARA BODOR II / ENGLISH	2	B12	11.00 - 13.00	BAI

Gambar 10. contoh pengujian hasil akhir dari program random penjadwalan kuliah

Dari output yang telah didapatkan kemudian dilakukan pengujian program. Tujuan dilakukan pengujian program yaitu untuk melihat kualitas dari output yang dihasilkan. Pengujian ini dilakukan menggunakan data STMIK Amik Riau.

C. Pembahasan

Hasil dari implementasi sistem penjadwalan *job shop* dalam perancangan jadwal kuliah menggunakan metode *constraint programming* adalah jumlah mata kuliah yang diinputkan pada sistem penjadwalan *job shop* menggunakan metode *constraints programming* adalah 114 mata kuliah, jumlah dosen yang diinputkan pada sistem penjadwalan *job shop* menggunakan metode *constraints programming* adalah 50 dosen, jumlah ruang yang diinputkan pada sistem penjadwalan *job shop* menggunakan metode *constraints programming* adalah 20 ruang serta jumlah slot/waktu yang diinputkan pada sistem penjadwalan *job shop* menggunakan metode *constraints programming* adalah 20 slot/waktu.

Dari hasil tersebut disimpulkan bahwa *penjadwalan job shop* dalam jadwal kuliah menggunakan metode *constraint programming* dapat mempermudah pihak jurusan/prodi dalam menyusun jadwal kuliah yang tepat waktu dan juga dapat menghindari terjadinya pelanggaran-pelanggaran yang mengakibatkan jadwal yang bentrok. Namun ada beberapa kelemahan yang didapatkan pada penelitian ini yaitu dosen tidak bisa melakukan request matakuliah, ruang, dan slot/waktu sesuai permintaan dosen tersebut sesudah penjadwalan tersusun oleh pihak jurusan/prodi ini dapat mengakibatkan terjadinya pelanggaran-pelanggaran pada penjadwalan kuliah.

IV. KESIMPULAN

Kesimpulan yang dapat diambil dari penelitian ini seperti yang telah diuraikan pada bab-bab yang sudah dibahas sebelumnya yaitu :

1. Penelitian ini menghasilkan suatu jadwal perkuliahan yang tidak melanggar *hard constraints*.
2. Jumlah output berupa jadwal kuliah yang memenuhi *hard constraints* yaitu 3 kali dari 10 kali pengujian *random* atau dengan persentase 30% jadwal *valid* dan hasil jadwal yang tidak *valid* sebanyak 7 kali dari total 10 kali *random* atau dengan persentase 75%.
3. Hasil perbandingan jadwal secara manual dengan jadwal menggunakan metode *constraints programming* mendapatkan hasil adanya penurunan pelanggaran *constraints*.
4. *Blind search* yang digunakan dalam penelitian ini sangat efisien dari segi waktu dikarenakan ruang solusi tidak besar walaupun dengan *blind search algorithm* ini prosesnya menguji seluruh ruang solusi yang ada.

REFERENSI

- [1] Adi Nugroho. (2005). "Analisa dan Perancangan Sistem Informatika dengan Metodologi Berorientasi Objek". Edisi Revisi Bandung : Penerbit Informatika.
- [2] Akerkar, Rajendra. (2005). "Introduction to artificial Intelligence". Halaman 122, Prentice-Hall, Hindia 2005.
- [3] Apt, K. (2005). *Explaining Constraint Programming*. Springer Berlin Heidelberg.
- [4] Betha Sidik, Ir., (2004). Pemrograman Web dengan PHP. Informatika, Bandung.
- [5] Djon Irwanto, S.Kom. MM. (2006) . "Perancangan Object Oriented Software dengan UML (Unified Modelling Language)". Ypgyakarta : Penerbit Andi Offset.
- [6] Dwi Prasetyo, Didik. 2003. Tip dan Trik Kolaborasi PHPdan MySQL untuk membuat web database yang interaktif, PT. Alex Media Komputindo, Jakarta.
- [7] Hutomo, A., Fitriandana, A., Marshadiany, A., Prikarti, G., & Imah, E. (2011). Implementasi Algoritma Integer Linier Programming Untuk Sistem Informasi Penjadwalan Ruang di Fakultas Ilmu Komputer Universitas Indonesia. *Jurnal of Information System*, 7 (1).
- [8] K. W, J., A., & Marsolim. (2006). Perancangan dan Implementasi Papan Jdwal Perkuliahan Berdasarkan Sistem Penjadwalan Otomatis. *Tesla*, 75-95.

-
- [9] Kusumadewi, Sri, “*Artificial Intelligence* “, halaman 23, Garaha Ilmu, Yogyakarta, 2003.
- [10] Lumbantoruan, R., Simatupang, Y. N., Siahaan, M. N., Pardede, M. H., & Pakpaham, J. (2012). Penjadwalan Kuliah dengan Algoritma Backtracking.
- [11] Mairiza, D. (n.d.). *Constraint Programming* : Suatu Pendekatan dalam Declarative Programming.
- [12] Mutakhiroh, I., Saptono, F., Hasanah, N., & Wiryadinata, R. (2007, Juni 16). Pemanfaatan Metode Heuristik dalam Pencarian Jalur Terpendek dengan Algoritma Semut dan Algoritma Genetika. SNATI, 1907-5022.
- [13] Naskhalim. (2014). *Constraint Programming* untuk Penjadwalan Job-Shop Para Guru. ISSN, III, 2339-210X.
- [14] Novriyanto, & Zaid. S, M. (n.d). Penerapan Algoritma *Bactracking* Berbasis *Blind Search* untuk menentukan Penjadwalan Mengajar
- [15] NST, & Anggi, M.R. (2010). Analisis dan Implementasi Algoritma Runut Balik (*Backtracking*) pada permainan Magin Squire.
- [16] Rochman, A. (2012, Juni 15-16). Penjadwalan kuliah menggunakan Metode Constraints Programming dan Simulated Annealing
- [17] Russell, S. J., & Norvig, P. (2003). *Artificial Intelligence A Modern Approach*. New Jersey.
- [18] Setiawan, N. (2007). PENENTUAN UKURAN SAMPEL MEMAKAI RUMUS.
- [19] Soraya, D. T. (2007). Penjadwalan Perkuliahan dengan pendekatan Constraint Programming. Univesitas Indonesia.
- [20] Stephen. (2014). Perbandingan Algoritma *Greedy & Backtracking* Dalam Penyelesaian Permainan 2048.
- [21] Sulun, H. S., & Munir, R. (2010). Pembangkit Teka-Teki Silang dengan Algoritma *Backtracking* serta Aplikasi Permainannya yang berbasis Web. Jurnal Informatika, 4.
- [22] Suyanto, “*Arificial Intelligence*”, halaman 15, Informatika, bandung, 2007.
- [23] Talbi, E. g. (2009). *Mataheuristik : from design to implementation*. Hoboken. New Jersey : A John Wiley & Sons, Inc.
- [24] Teddy. (2009). Penyelesaian Penjadwalan Kuliah sebagai Constraints Satisfaction Problem Dengan Genetika Algorithm. Depok: Fasilkom UI