UNIVERSITAS INDONESIA

PENDEKATAN *COLUMN GENERATION* PADA MASALAH PENJADWALAN MATA KULIAH

SKRIPSI

SUTISNA
0606067856

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
PROGRAM STUDI SARJANA MATEMATIKA
DEPOK
JULI 2011

UNIVERSITAS INDONESIA

PENDEKATAN *COLUMN GENERATION* PADA MASALAH PENJADWALAN MATA KULIAH

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar sarjana sains

SUTISNA
0606067856

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
PROGRAM STUDI SARJANA MATEMATIKA
DEPOK

JULI 2011

HALAMAN PERNYATAAN ORISINALITAS

Skripsi ini adalah hasil karya sendiri,

dan semua sumber baik yang dikutip maupun dirujuk

telah saya nyatakan dengan benar.

Nama	: Sutisna
NPM	: 0606067856
Tanda Tangan	: 
Tanggal	: 12 Juli 2011

HALAMAN PENGESAHAN

Skripsi ini diajukan oleh
Nama : Sutisna
NPM : 0606067856
Program Studi : Sarjana Matematika
Judul Skripsi : Pendekatan *Column Generation* pada Masalah Penjadwalan Mata Kuliah

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Sains pada Program Studi S1 Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Indonesia

DEWAN PENGUJI

Pembimbing I	:	Rahmi Rusin, M.ScTech.	( )
Pembimbing II	:	Dr. Yudi Satria, M.T.	( )
Penguji	:	Dra. Denny Riama Silaban, M.Kom.	( )
Penguji	:	Helen Burhan, M.Si.	( )
Penguji	:	Dhian Widya, M.Kom.	( )

Ditetapkan di : Depok
Tanggal : 14 Juni 2011

KATA PENGANTAR

Alhamdulillah, puji syukur kepada Allah SWT. atas semua rahmat dan karunia yang telah Dia berikan sehingga penulis dapat menyelesaikan tugas akhir ini. Penulis sadar bahwa penyelesaian tugas akhir ini tidak terlepas dari bantuan dan dukungan dari berbagai pihak. Oleh karena itu, pada kesempatan ini penulis ingin mengucapkan terima kasih kepada pihak-pihak yang telah berjasa dalam penulisan tugas akhir ini maupun selama penulis kuliah. Ucapan terima kasih terhatur kepada:

(1) Ibu Rahmi Rusin M.Sc.Tech selaku pembimbing pertama dan Bapak Dr Yudi Satria, M.T selaku pembimbing kedua. Terima kasih sebesar-besarnya untuk semua bantuan, saran, kritik dan masukan yang diberikan kepada penulis dalam menyelesaikan tugas akhir ini.

(2) Ibu Dr. Kiki A. Sugeng selaku pembimbing akademik penulis selama menjalani masa kuliah. Terima kasih atas bimbingan yang terus menerus Ibu berikan selama 5 tahun ini.

(3) Seluruh staf pengajar di departemen Matematika UI, terima kasih atas segala ilmu yang telah diberikan. Semoga penulis dapat menggunakan ilmu tersebut dengan sebaik-baiknya.

(4) Seluruh karyawan di departemen Matematika UI, terima kasih atas segala bantuan dan kemudahan yang telah diberikan untuk penulis.

(5) Bapak, Ibu, Kakak, dan Adik penulis yang selalu memberikan doa, semangat, dan dukungan bagi penulis. Terima kasih pula karena telah mendukung semua keinginan dan harapan penulis.

(6) Teman-teman yang sering duduk di depan gedung matematika yang telah menjadi teman tertawa dan melepas penat bersama.

(7) Teman-teman pengurus HMD matematika 2007/2008 atas kerja sama dan persaudaraan yang terjalin selama ini.

(8) Teman-teman 2006: Anggha, Michael, Oza, Aliman, Bara, Syafira, Lee, Yuri, Rita, Cims, Rendi, Rian, Billy, Oppie, Mei, Inne, Rizkyatul, Rahanti, Stefani,

Faal, Mella, Milla, Tami, Annisa, Widya, Latief, Hot, Noor, Farah, Putri Helmet, Purwita, Nurgi, Lena, Nadia, Reza, Rifza, Yunita, Indra, Puspa, Febrian, Poe, Tasya, Billy, Rahmanita, Kiki, Nobo, Tika, Rontu, Lani, Dita, Budi, Rama, Teguh, Dodi, Rafly, Ali, Pangky, Rita, untuk kebersamaan yang telah dijalani.

(9) Teman-teman BALITA (barisan lima tahun) yang menjadi teman senasib.

(10) Teman-teman angkatan 2007, 2008, 2009 dan 2010.

(11) Seluruh teman mahasiswa matematika UI.

Penulis juga ingin mengucapkan terima kasih kepada seluruh pihak yang tidak dapat disebutkan satu per satu, yang telah membantu dalam penyusunan skripsi ini. Akhir kata, penulis mohon maaf jika terdapat kesalahan atau kekurangan dalam skripsi ini. Penulis berharap semoga skripsi ini bermanfaat bagi pembaca.

Penulis

2011

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini:

Nama	:	Sutisna
NPM	:	0606067856
Program Studi	:	Sarjana Matematika
Departemen	:	Matematika
Fakultas	:	Matematika dan Ilmu Pengetahuan Alam
Jenis karya	:	Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty Free Right*) atas karya ilmiah saya yang berjudul :

Pendekatan *Column Generation* pada Masalah Penjadwalan Mata Kuliah

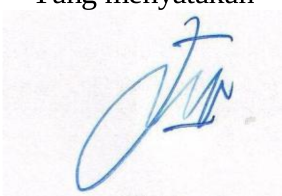
beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/format-kan, mengelola dalam bentuk pangkalan data (database), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok

Pada tanggal : 12 Juli 2011

Yang menyatakan



(Sutisna)

ABSTRAK

Nama : Sutisna
Program Studi : Matematika
Judul : Pendekatan *Column Generation* pada Masalah Penjadwalan Mata Kuliah

Dalam setiap semester, setiap jurusan di universitas menghadapi permasalahan yang sama yaitu menjadwalkan mata kuliah dengan waktu dan ruangan tertentu dimana terdapat beberapa batasan atau kendala. Dalam penjadwalan, mata kuliah harus dijadwalkan dalam waktu dan ruangan tertentu, dimana tidak terdapat mata kuliah di waktu yang sama diajarkan di ruangan yang sama dan tidak boleh bersamaan waktu antara mata kuliah dalam kelompok yang sama. Diberikan pemilihan waktu oleh dosen untuk mata kuliah yang diajarkannya, masalah penjadwalan mata kuliah diformulasikan sebagai pemrograman bilangan bulat dengan fungsi tujuan adalah memaksimumkan pemilihan waktu oleh dosen, dimana metode *Column Generation* digunakan untuk mencari solusi optimal dari model relaksasinya. Setiap kolom merepresentasikan pola jadwal mingguan dari tiap mata kuliah. Kolom akan dibangkitkan untuk mendapatkan pemilihan waktu terbaik dalam seminggu. Solusi optimal didapatkan ketika tidak ada lagi kolom yang dibangkitkan dan solusi memenuhi kondisi integral. Pada skripsi ini masalah penjadwalan mata kuliah diaplikasikan pada Departemen Matematika UI untuk perkuliahan di semester genap. Pembuatan program untuk menyelesaikan masalah penjadwalan mata kuliah menggunakan perangkat lunak dan hasilnya didapatkan solusi optimal yang memenuhi seluruh kendala.

Kata Kunci : *Column Generation*, *Branch and Price*, Penjadwalan Mata Kuliah, Pemrograman Linier Bilangan Bulat
xii+56 halaman : 3 gambar + 4 tabel
Daftar Pustaka : 9 (1981-2007)

ABSTRACT

Name : Sutisna
Study Program : Mathematics
Title : Column Generation Approach to Course Scheduling Problem

In each semester, every department in the university faces the same problem of courses scheduling in a certain time and classroom with some constraints. In scheduling, the courses must be scheduled, where there are no subjects put at the same time in the same room and it is not allowed to overlap between subjects in the same group. Given the preferences of the lecturers to teaching time, a course scheduling problem is formulated as an integer programming with objective function is to maximize the preference value of the lecturers. The column generation approach is used to find the optimal solution of the relaxation model. Each column represents a pattern of weekly schedule of each course. The column will be generated to get the best solution. The optimal solution is obtained when no more column is generated and the solution satisfies the integral condition. In this *skripsi*, the column generation approach is applied to scheduling problem at Department of Mathematics UI for courses in second term each year. A program made to solve scheduling problems using a software and the obtained solution is satisfying all constraints.

Keywords : Column Generation, Branch and Price, Course Scheduling, Integer Linier Programming
xii+56 pages : 3 figures + 4 tables
Bibliography : 9 (1981-2007)

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB 1
PENDAHULUAN

1.1 Latar Belakang

Masalah penjadwalan merupakan bentuk umum suatu percobaan untuk mengidentifikasi keoptimalan dari aktivitas-aktivitas pada jam-jam tertentu setiap harinya dan beberapa hari dalam periode tertentu, biasanya satu minggu, (Papoutsis ,K., dkk., 2003). Masalah ini muncul dalam berbagai bidang, contohnya dalam bidang pendidikan. Dalam setiap semester, setiap jurusan di universitas menghadapi permasalahan yang sama yaitu menjadwalkan mata kuliah dengan waktu dan ruangan tertentu dimana terdapat beberapa batasan atau kendala.

Masalah penjadwalan di universitas dapat diselesaikan dalam banyak cara yang berbeda. Beberapa diantaranya yaitu penjadwalan mata kuliah untuk Departemen Matematika UI dengan algoritma memetika (Lismanto, 2008) dan penjadwalan ujian dengan pewarnaan graf (Sitorus, L., 2011). Dalam skripsi ini akan dibahas penjadwalan mata kuliah dengan pendekatan pembangkitan kolom (*column generation*). Metode ini telah sukses digunakan untuk penyelesaian beberapa optimisasi kombinatorik dan masalah penjadwalan (Papoutsis, K., dkk., 2003). Metode *column generation* pertama kali diperkenalkan oleh Gilmore dan Gomory pada tahun 1961 untuk menyelesaikan masalah *cutting stock problem* satu dimensi yang merupakan masalah program linier.

Dalam penjadwalan mata kuliah, masalah yang dihadapi adalah bagaimana menempatkan mata kuliah pada waktu dan ruangan tertentu sehingga didapatkan hasil yang optimal. Mata kuliah harus dijadwalkan dalam waktu dan ruangan tertentu, dimana tidak terdapat dua mata kuliah atau lebih yang diajarkan di waktu yang sama ditempatkan di ruangan yang sama dan juga tidak boleh bersamaan waktu antara mata kuliah-mata kuliah dalam kelompok yang sama.

Dalam hal ini kelompok yang sama adalah himpunan mata kuliah-mata kuliah yang diajarkan oleh dosen yang sama atau kelompok mata kuliah-mata kuliah yang diajarkan untuk mahasiswa yang sejenis. Kelompok mahasiswa sejenis misalkan mahasiswa tingkat 1, tingkat 2, kelompok mahasiswa dengan peminatan sama dan seterusnya. Pada penjadwalan mata kuliah, dosen dibolehkan untuk memilih waktu pengajaran untuk mata kuliah yang diampunya. Namun, pemilihan tersebut tidak boleh sembarangan dan harus diketahui oleh pembuat jadwal. Pemilihan waktu inilah yang akan dioptimalkan dalam masalah penjadwalan mata kuliah.

Untuk mengukur kualitas baik atau tidaknya suatu penjadwalan, dapat dilihat dari terpenuhinya semaksimal mungkin pemilihan waktu oleh dosen dan ruangan serta kendala-kendala lain sesuai dengan kondisi yang ada. Permasalahan penjadwalan ini dapat dimodelkan dengan menggunakan pemrograman linier bilangan bulat biner. Pada model tersebut, variabel yang digunakan berasosiasi dengan dipakai atau tidaknya pola penjadwalan mata kuliah tertentu untuk ditempatkan di ruangan dan waktu tertentu (Papoutsis, K., dkk., 2003). Banyaknya pola penjadwalan adalah kombinasi dari banyaknya mata kuliah dan banyaknya waktu.

Pada beberapa masalah program linier, Setiap variabel bersesuaian dengan kolom dalam matriks kendalanya. Sehingga jika variabel sangat banyak identik dengan melibatkan kolom yang sangat besar. Ide dari *column generation* cukup mengambil subhimpunan dari himpunan kolom yang besar untuk diselesaikan. Kemudian kolom baru dibangkitkan hanya saat diperlukan, yaitu ketika variabel yang bersesuaian dengan kolom tersebut berpotensi mengoptimalkan fungsi tujuan.

Karena kemungkinan pola penjadwalan setiap mata kuliah yang jumlahnya sangat banyak, maka metode simpleks yang biasa digunakan untuk menyelesaikan masalah program linier tidak mungkin digunakan. Oleh karena itu, akan digunakan pendekatan *column generation* mendapatkan penjadwalan kuliah yang paling optimal.

1.2 Perumusan Masalah dan Ruang Lingkup

Masalah yang akan dibahas dalam skripsi ini adalah:

1. Bagaimanakah pendekatan *column generation* dalam penyelesaian masalah penjadwalan mata kuliah sehingga diperoleh jadwal yang optimal.
2. Bagaimana membuat jadwal kuliah yang optimal di Departemen Matematika Universitas Indonesia program studi S1.
3. Bagaimana pembuatan program masalah penjadwalan mata kuliah dengan bantuan software.

1.3 Jenis Penelitian dan Metode yang Digunakan

Jenis penelitian yang digunakan adalah studi literatur dan metode yang digunakan adalah pengkonstruksian penjadwalan mata kuliah dengan pendekatan *column generation*.

1.4 Tujuan

Tujuan dari penulisan skripsi ini adalah:

1. Menjelaskan pendekatan *column generation* dalam menyelesaikan masalah penjadwalan mata kuliah di universitas.
2. Mengkonstruksi penjadwalan mata kuliah yang optimal di Departemen Matematika UI program studi S1 dengan kendala-kendala yang ada.
3. Membuat program masalah penjadwalan mata kuliah dengan bantuan software.

BAB 2
LANDASAN TEORI

Pada bab ini akan dibahas landasan teori yang digunakan pada bab selanjutnya, yaitu masalah program linier, himpunan konveks, metode simpleks, masalah dual, masalah pemrograman bilangan bulat, dan *branch and price* yang merupakan gabungan antara metode *branch and bound* dan *column generation*. Pembahasan bab ini merujuk pada jurnal Lubbecke, M.E dan J. Desrosiers, 2002, buku Hillier, F.S dan Lieberman, G.J, jurnal Barnhart C., E.L. Johnson, G.L. Nemhauser, M.W.P. Savelsbergh dan P. Vance, 1988, buku Wu, N., dan Coppins, R, 1981, tesis Burhan, H, dan skripsi Mahayekti, H, 2007.

2.1 Masalah Program Linier

Secara umum masalah program linier dapat didefinisikan sebagai masalah mengoptimalkan (memaksimumkan atau meminimumkan) suatu fungsi linier dalam variabel x_j, $j = 1,2,3, \dots, n$ terhadap sejumlah berhingga kendala linear. Fungsi linier yang dioptimalkan disebut fungsi tujuan dan kendala linier dapat berupa persamaan linier dan pertidaksamaan linier.

Bentuk umum masalah program linier untuk masalah meminimumkan dapat dinyatakan dalam bentuk matematis berikut ini:

$$\text{minimum} \qquad f = \sum_{j \in J} c_j x_j$$

$$\text{dengan kendala (dk)} \qquad \sum_{j \in J} a_{ij}\, x_j \geq b_i, i = 1,2,3, \dots, m$$

$$x_j \geq 0, \;\; j = 1,2,3, \dots, n$$

Untuk kendala yang berupa pertidaksamaan $\leq$ kedua ruas kendala dikalikan dengan -1.

Setiap masalah program linier dapat diubah ke dalam bentuk baku. Suatu program linier dikatakan dalam bentuk baku jika semua kendalanya berupa persamaan dan semua variabel bernilai nonnegatif. Langkah-langkah dalam mengubah suatu bentuk umum masalah program linier ke dalam bentuk baku adalah sebagai berikut:

1. Jika kendala ke-i berupa kendala pertidaksamaan $\leq$, maka kendala tersebut diubah menjadi bentuk persamaan dengan menambah *slack variable*, s_i, pada ruas kiri kendala ke-i, dengan $s_i \geq 0$, sedangkan pada fungsi tujuan ditambahkan dengan $0s_i$.
2. Jika kendala ke-i berupa kendala pertidaksamaan $\geq$, maka kendala tersebut diubah menjadi bentuk persamaan dengan menambah $-e_i$, dimana e_i merupakan *excess variable*, pada ruas kiri kendala ke-i, dengan $e_i \geq 0$,dan pada fungsi tujuan ditambahkan dengan $0e_i$.

Secara matematis bentuk baku dari program linier dapat dinyatakan sebagai berikut:

$$\text{minimum} \quad f = \boldsymbol{c}^{\boldsymbol{T}}\mathbf{x} \tag{2.1}$$

$$\text{dk} \quad A\mathbf{x} = \mathbf{b} \tag{2.2}$$

$$\mathbf{x} \geq \mathbf{0} \tag{2.3}$$

Dengan $\mathbf{x}$ adalah matriks berukuran $n \times 1$, matriks A berukuran $m \times n$, dan matriks $\mathbf{b}$ berukuran $m \times 1$, vektor $\boldsymbol{c}^{\boldsymbol{T}}$ adalah vektor dengan entri koefisien fungsi tujuan berukuran $n \times 1$.

Dapat dilihat bahwa solusi optimal dari bentuk baku masalah program linier di atas bergantung pada konsistensi dari kendala (2.2) yang merupakan sistem persamaan linier, berikut ini akan dibahas solusi basis yang merupakan dasar untuk mendapatkan solusi optimal dari masalah program linier.

2.2 Solusi Basis

Pandang sistem persamaan linier (2.2) dengan m persamaan dan n variabel (asumsikan $m < n$). Vektor $\mathbf{x}$ tak negatif yang memenuhi kendala (2.2) disebut sebagai solusi basis *feasible*. Solusi basis dari $A\mathbf{x} = \mathbf{b}$ adalah solusi dimana terdapat sebanyak-banyaknya m variabel bernilai tidak nol. Untuk mendapatkan solusi basis tersebut, maka sebanyak $n - m$ variabel harus bernilai nol. Variabel yang bernilai nol ini disebut *variabel non basis*, sedangkan variabel yang bernilai tidak nol disebut *variabel basis*.

Berikut ini langkah-langkah untuk mencari solusi basis dari $A\mathbf{x} = \mathbf{b}$:

- Langkah pertama adalah mempartisi matriks A menjadi $A^{'} = [B|N]$, $B \in M(m, m)$ dan $N \in M(m, n - m)$. B adalah matriks persegi yang berisi kolom-kolom yang bebas linier di A, sehingga B invertibel.
- Seiring dengan partisi dari A maka vektor variabel $\mathbf{x}$ di partisi menjadi $\mathbf{x}' = \binom{\mathbf{x}_B}{\mathbf{x}_N}$ dengan $\mathbf{x}_B$ disebut variabel basis dan $\mathbf{x}_N$ disebut variabel non basis.
- Sehingga $A\mathbf{x} = \mathbf{b}$ dapat dinyatakan dengan

$$(B|N)\binom{\mathbf{x}_B}{\mathbf{x}_N} = \mathbf{b}.$$

$$B\mathbf{x}_B + N\mathbf{x}_N = \mathbf{b}$$

$$B\mathbf{x}_B = \mathbf{b} - N\mathbf{x}_N \tag{2.4}$$

- Vektor $\mathbf{x}_B$ dapat diperoleh dari (2.4), yaitu dengan mengalikan kedua ruas (2.4) dengan B^{-1}, sehingga didapat :

$$B^{-1}B\mathbf{x}_B = B^{-1}\mathbf{b} - B^{-1}N\mathbf{x}_N$$

$$\mathbf{x}_B = B^{-1}\mathbf{b} - B^{-1}N\mathbf{x}_N \tag{2.5}$$

- Jika $\mathbf{x}_N = 0$ didapat $\mathbf{x}_B = B^{-1}\mathbf{b}$, sehingga solusi basis $A\mathbf{x} = \mathbf{b}$ adalah $\mathbf{x} = \binom{B^{-1}\mathbf{b}}{0}$.

Karena vektor $\mathbf{x}$ menjadi vektor $\mathbf{x}' = \binom{\mathbf{x}_B}{\mathbf{x}_N}$, maka vektor koefisien fungsi tujuan $\boldsymbol{c}^{\boldsymbol{T}}$ juga dapat dipartisi menjadi $\boldsymbol{c}'^{\boldsymbol{T}} = [\boldsymbol{c}_{\boldsymbol{B}}^{\boldsymbol{T}}|\boldsymbol{c}_{\boldsymbol{N}}^{\boldsymbol{T}}]$. dimana $\boldsymbol{c}_{\boldsymbol{B}}^{\boldsymbol{T}}$ merupakan vektor koefisien fungsi tujuan yang bersesuaian dengan vektor $\mathbf{x}_B$ dan $\boldsymbol{c}_{\boldsymbol{N}}^{\boldsymbol{T}}$ vektor koefisien fungsi tujuan yang bersesuaian dengan vektor $\mathbf{x}_N$. Sehingga fungsi tujuan (2.5) dapat ditulis menjadi:

$$f = [\boldsymbol{c}_{\boldsymbol{B}}^{\boldsymbol{T}}|\boldsymbol{c}_{\boldsymbol{N}}^{\boldsymbol{T}}]\begin{pmatrix}\mathbf{x}_B\\ \mathbf{x}_N\end{pmatrix} \tag{2.6}$$

dengan mensubstitusikan persamaan (2.5) ke (2.6) akan didapat:

$$\begin{aligned} f &= \boldsymbol{c_B^T}\mathbf{x}_B + \boldsymbol{c_N^T}\mathbf{x}_N \\ &= \boldsymbol{c_B^T}(B^{-1}\mathbf{b} - B^{-1}N\mathbf{x}_N) + \boldsymbol{c_N^T}\mathbf{x}_N \\ &= \boldsymbol{c_B^T}B^{-1}\mathbf{b} + (\boldsymbol{c_N^T} - \boldsymbol{c_B^T}B^{-1}N)\mathbf{x}_N \end{aligned}$$

Koefisien $\boldsymbol{r_N} = \boldsymbol{c_N^T} - \boldsymbol{c_B^T}B^{-1}N$ disebut *reduced cost vector* dari variabel non basis $\mathbf{x}_N$. Jika nilai $\boldsymbol{r_N} < 0$, maka nilai f akan berkurang, sehingga solusi basis *feasibel* dari masalah program linier (2.1)-(2.3) dikatakan optimal jika $\boldsymbol{r_N} \geq 0$.

2.3 Himpunan Konveks

Sebelumnya telah dibahas masalah program linier dari sudut pandang aljabar. Berikut ini akan dibahas mengenai teori himpunan konveks yang merupakan konsep dasar dalam menyelesaikan masalah program linier dari sudut pandang geometri.

Pandang suatu program linier dari (2.1)-(2.3). Misalkan R^n adalah suatu ruang hasil kali dalam Euclid. Berikut akan diberikan, definisi dan teorema yang dikutip dari (Luenberger, David G., 1973).

Definisi 2.3.1 Suatu $C \subseteq R^n$ disebut konveks, jika untuk setiap $\mathbf{x}, \mathbf{y} \in C$ dan $0 \leq \lambda \leq 1$ berlaku $\lambda\mathbf{x} + (1-\lambda)\mathbf{y} \in C$.

Dari definisi di atas dikatakan bahwa suatu himpunan C yang merupakan himpunan bagian dari R^n disebut konveks jika setiap segmen garis yang menghubungkan sembarang dua buah titik elemen C terdapat juga di C.

Teorema 2.3.2 Jika F suatu keluarga dari subhimpunan konveks dari R^n maka $D = \bigcap\{C | C \in F\}$ adalah konveks.

Bukti: Jika $D = \emptyset$ maka dari Definisi 2.3.1 jelas bahwa D konveks. Misalkan $D \neq \emptyset$. Jika $\mathbf{x}, \mathbf{y} \in D$ dan $0 \leq \lambda \leq 1$, maka $\mathbf{x}, \mathbf{y} \in C$ untuk setiap $C \subseteq F$. Karena C konveks maka berlaku $\mathbf{z} = \lambda\mathbf{x} + (1-\lambda)\mathbf{y} \in C$ untuk setiap $C \subseteq F$, terbukti.

Kendala (2.2) mempunyai intrepretasi geometri yang dapat dinyatakan dalam definisi berikut ini.

Definisi 2.3.3 Jika $\mathbf{a} \in R^n$ dengan $\mathbf{a} \neq 0$, dan $b \in R$, maka himpunan $\{\mathbf{x} \in R^n | \langle \mathbf{a}, \mathbf{x} \rangle = b\}$ disebut **hyperplane** di R^n. Dengan $\langle \mathbf{a}, \mathbf{x} \rangle = a_1x_1 + a_2x_2 + \cdots + a_nx_n$ merupakan hasil kali dalam Euclid di R^n.

Definisi 2.3.4 Himpunan $\{\mathbf{x} \in R^n | \langle \mathbf{a}, \mathbf{x} \rangle \leq b\}$ disebut *closed half-space* di R^n, sedangkan himpunan $\{\mathbf{x} \in R^n | \langle \mathbf{a}, \mathbf{x} \rangle < b\}$ disebut *open half-space* di R^n.

Teorema 2.3.5 Setiap *half-space* di R^n adalah konveks.
Bukti: Tanpa mengurangi keumuman, misalkan **x** dan **y** anggota dari *closed half-space* $\{\mathbf{x} \in R^n | \langle \mathbf{a}, \mathbf{x} \rangle \leq b\}$. Akan dibuktikan $\lambda\mathbf{x} + (1-\lambda)\mathbf{y} \in \{\mathbf{x} \in R^n | \langle \mathbf{a}, \mathbf{x} \rangle \leq b\}$, yaitu akan ditunjukkan $\langle \mathbf{a}, \lambda\mathbf{x} + (1-\lambda)\mathbf{y} \rangle \leq b$. Berdasarkan definisi hasil kali dalam Euclid di R^n, maka $\langle \mathbf{a}, \lambda\mathbf{x} + (1-\lambda)\mathbf{y} \rangle = \lambda\langle \mathbf{a}, \mathbf{x} \rangle + (1-\lambda)\langle \mathbf{a}, \mathbf{y} \rangle$ dengan $\lambda \geq 0$ dan $(1-\lambda) \geq 0$. Karena $\mathbf{x}, \mathbf{y} \in \{\mathbf{x} \in R^n | \langle \mathbf{a}, \mathbf{x} \rangle \leq b\}$, maka $\langle \mathbf{a}, \mathbf{x} \rangle \leq b$ dan $\langle \mathbf{a}, \mathbf{y} \rangle \leq b$. Sehingga

$$\begin{aligned}\langle \mathbf{a}, \lambda\mathbf{x} + (1-\lambda)\mathbf{y} \rangle &= \lambda\langle \mathbf{a}, \mathbf{x} \rangle + (1-\lambda)\langle \mathbf{a}, \mathbf{y} \rangle \\ &\leq \lambda b + (1-\lambda)b \\ &\leq b\end{aligned}$$

Akibat 2.3.6 Setiap **hyperplane** di R^n adalah konveks.
Bukti: Suatu *hyperplane* $\{\mathbf{x} \in R^n | \langle \mathbf{a}, \mathbf{x} \rangle = b\}$ adalah irisan dari *closed half-space* $\{\mathbf{x} \in R^n | \langle \mathbf{a}, \mathbf{x} \rangle \leq b\}$ dan $\{\mathbf{x} \in R^n | \langle \mathbf{a}, \mathbf{x} \rangle \geq b\}$ yang merupakan himpunan konveks. Sehingga berdasarkan Teorema 2.3.2, *hyperplane* adalah konveks.

Intrepretasi geometri untuk kendala (2.3) dinyatakan dalam Akibat 2.3.7 berikut ini.

Akibat 2.3.7 Himpunan $\{\mathbf{x} \in R^n | x_i \geq 0, i = 1,2, \ldots n\}$ adalah konveks.
Bukti: $x_i \geq 0$ jika dan hanya jika $\langle \mathbf{e_i}, \mathbf{x} \rangle \geq 0$, dimana $\mathbf{e_i}$ adalah vektor satuan standar di R^n. Himpunan $\{\mathbf{x} \in R^n | x_i \geq 0, i = 1,2, \ldots, n\}$ merupakan irisan dari n *half-space* $\{\mathbf{x} \in R^n | \langle \mathbf{e_i}, \mathbf{x} \rangle \geq 0\}$. Berdasarkan Teorema 2.3.5 dan 2.3.2, maka himpunan $\{\mathbf{x} \in R^n | x_i \geq 0, i = 1,2, \ldots, n\}$ adalah konveks.

Misalkan $S \subseteq R^n$ dan F adalah keluarga dari semua himpunan konveks di R^n yang memuat S. Jelas bahwa $F \neq \phi$, karena $R^n \in F$.

Definisi 2.3.8 Himpunan konveks terkecil yang memuat S, $\langle S \rangle = \bigcap\{C | C \in F\}$ disebut ***convex hull*** dari S.

Definisi 2.3.9 Suatu himpunan konveks C yang terbatas disebut ***polytope***, dimana $C = \langle S \rangle$ untuk suatu subhimpunan hingga S.

Berdasarkan definisi, teorema, dan akibat di atas didapat bahwa daerah *feasible* dari suatu masalah program linier (2.1)-(2.3) merupakan suatu himpunan konveks yang disebut sebagai ***polytope***.

Sebelum membahas solusi optimal masalah program linier dengan daerah *feasible*-nya berupa suatu ***polytope***, berikut akan dijelaskan mengenai kombinasi konveks.

Teorema 2.3.10 Misalkan C adalah subhimpunan konveks di R^n. Jika $x_i \in C$, $\lambda_i \geq 0$, $i = 1,2,\dots,k$, dan $\sum_{i=1}^{k} \lambda_i = 1$ maka $\sum_{i=1}^{k} \lambda_i\, x_i$ ada di C. $\sum_{i=1}^{k} \lambda_i\, x_i$ disebut kombinasi konveks dari x_i.

Bukti:

Akan dibuktikan dengan menggunakan induksi terhadap k.

Jika $k = 1$ maka diketahui $x_1 \in C$, $\lambda_1 \geq 0$ dan $\sum_{i=1}^{1} \lambda_1 = 1$ sehingga $\sum_{i=1}^{1} \lambda_i\, x_i = 1.x_1 = x_1 \in C$, jelas teorema berlaku.

Asumsikan pernyataan berlaku hingga $k = m$, sehingga $\sum_{i=1}^{m} \lambda_i\, x_i \in C$. Akan dibuktikan pernyataan berlaku untuk $k = m + 1$.

$$\sum_{i=1}^{m+1} \lambda_i\, x_i = \sum_{i=1}^{m} \lambda_i\, x_i + \lambda_{m+1} x_{m+1} = (1 - \lambda_{m+1}) \sum_{i=1}^{m} \mu_i\, x_i + \lambda_{m+1} x_{m+1} \quad (2.7)$$

dimana $\mu_i = \frac{\lambda_i}{(1-\lambda_{m+1})}, i = 1,2,\dots,m$

Karena $\lambda_i \geq 0, i = 1,2,\dots,m+1$ dan $\sum_{i=1}^{m} \lambda_i = 1$ maka didapat $\mu_i \geq 0, i = 1,2,\dots,m$ dan

$$\sum_{i=1}^{m} \mu_i = \frac{\sum_{i=1}^{m} \lambda_i}{(1 - \lambda_{m+1})} = \frac{(1 - \lambda_{m+1})}{(1 - \lambda_{m+1})} = 1$$

Berdasarkan asumsi untuk $k = m$, misalkan $y = \sum_{i=1}^{m} \mu_i \, x_i \in C$, sehingga persamaan (2.7) dapat ditulis sebagai berikut:

$$\sum_{i=1}^{m+1} \lambda_i \, x_i = (1 - \lambda_{m+1})y + \lambda_{m+1} x_i$$

Berdasarkan sifat konveks C maka $\sum_{i=1}^{m+1} \lambda_i \, x_i \in C$.

Berikut akan dijelaskan tentang nilai maksimum atau nilai minimum dari masalah program linier yang didefinisikan pada suatu ***polytope*** terjadi pada titik ekstrem di ***polytope*** tersebut.

Definisi 2.3.11 Misalkan $C \neq \emptyset$ dan $C \subseteq R^n$. Maka **x** disebut titik ekstrem dari C jika **x** hanya dapat dibuat sebagai kombinasi konveks dari **x** itu sendiri.

Teorema 2.3.12 Jika C adalah suatu **polytope** di R^n dan f adalah suatu fungsi linear di R^n maka $\min_{\mathbf{x} \in C} f$ dan $\max_{\mathbf{x} \in C} f$ ada dan terjadi di titik ekstrem dari C.

Sehingga dapat disimpulkan bahwa nilai fungsi tujuan (2.1) terjadi pada titik ekstrem dari suatu **polytope** yang merupakan daerah *feasible*.

2.4 Metode Simpleks

Metode simpleks merupakan salah satu cara yang sering digunakan untuk menyelesaikan masalah program linier. Metode simpleks merupakan proses aljabar yang bersifat iteratif. Pada setiap iterasi dalam metode simpleks, satu variabel nonbasis akan masuk menjadi variabel basis dan satu variabel basis akan keluar menjadi variabel nonbasis. Variabel nonbasis yang masuk menjadi variabel basis disebut variabel masuk dan variabel basis yang keluar menjadi variabel non basis disebut variabel keluar. Dalam kasus meminimumkan, yang dipilih sebagai variabel masuk adalah variabel nonbasis yang mempunyai nilai *reduced cost* yang paling negatif.

Berikut akan dijelaskan algoritma metode simpleks:

1. Konversikan masalah program linier ke dalam bentuk baku.

2. Tentukan solusi basis *feasible* dari $A\mathbf{x} = \mathbf{b}$
3. Hitung *reduced cost* setiap variabel nonbasis j, yaitu $\mathbf{r_j} = c_j - \mathbf{c_B^T} a_j$. Jika *reduced cost* setiap variabel nonbasis, $\mathbf{r_N} \geq 0$, maka solusi basis *feasible* merupakan solusi optimal, jika $\mathbf{r_N} < 0$ maka pilih *reduced cost* variabel nonbasis yang paling negatif, misalkan $\mathrm{x_t}$.
4. Jika terdapat *reduced cost* yang dapat mengubah nilai fungsi tujuan menjadi lebih optimal, maka masukkan variabel yang bersesuaian dengan *reduced cost* tersebut untuk menjadi variabel basis, lalu carilah variabel yang akan keluar dari basis.
5. Lakukan operasi baris dasar untuk mencari solusi basis *feasible* baru, dan *reduced cost* setiap variabel nonbasis.
6. Ulangi langkah 3 hingga didapat solusi optimal.

2.5 Masalah Dual

Setiap masalah program linier (disebut masalah primal) dalam bentuk baku berasosiasi dengan masalah program linier lain, yang disebut sebagai masalah dual. Masalah primal dan dual ini saling berhubungan, yaitu setiap solusi *feasible* dari masalah yang satu menghasilkan suatu batas dari nilai optimal masalah yang lainnya. Masalah primal dalam bentuk umum masalah meminimumkan adalah sebagai berikut :

$$\begin{aligned} \text{minimum} \quad & f = \mathbf{c^T x} \\ \text{dk} \quad & A\mathbf{x} \geq \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0} \end{aligned} \tag{2.7}$$

Masalah dual dari masalah primal di atas adalah :

$$\begin{aligned} \text{maksimum} \quad & g = \mathbf{y^T b} \\ \text{dk} \quad & \mathbf{y^T} A \leq \mathbf{c^T} \\ & \mathbf{y} \geq \mathbf{0} \end{aligned} \tag{2.8}$$

Berdasarkan model di atas didapat hubungan antara masalah primal dan dual dari suatu program linier (Wu, N., dan Coppins, R., 1981), sebagai berikut :

1. Sifat dualitas lemah/*Weak Duality property* : Jika $\bar{\mathbf{x}}$ adalah solusi *feasible* untuk masalah primal (2.7) dan $\bar{\mathbf{y}}$ adalah solusi *feasible* dari masalah dual (2.8), maka: $\mathbf{y}^{\mathbf{T}}\mathbf{b} \leq \mathbf{c}^{\mathbf{T}}\mathbf{x}$.
2. Sifat dualitas kuat/*Strong Duality property* : Jika $\bar{\mathbf{x}}$ adalah solusi *feasible* dari masalah primal (2.7) dan $\bar{\mathbf{y}}$ adalah solusi optimal dari masalah dual (2.8), dan jika $\mathbf{y}^{\mathbf{T}}\mathbf{b} = \mathbf{c}^{\mathbf{T}}\mathbf{x}$, maka $\bar{\mathbf{x}}$ dan $\bar{\mathbf{y}}$ masing-masing merupakan solusi optimal untuk masalah (2.7) dan (2.8).

2.6 *Integer Linier Programming*

Masalah program linier dimana nilai semua variabelnya bernilai bilangan bulat disebut masalah pemrograman linier bilangan bulat murni (*integer linier programming*). Dan jika nilai dari variabel adalah 0 atau 1, masalah ini disebut pemrograman bilangan bulat biner (*binary integer programming*). Secara matematis dapat dinyatakan sebagai berikut :

$$\begin{array}{ll} \text{maksimum} & z = \sum_{j \in J} c_j \, x_j \\ \text{dk} & \sum_{j \in J} a_{ij} \, x_j \leq b_i, \quad i = 1,2,3,\dots,m \\ & x_j \geq 0, \qquad j \in J \\ & x_j \in \mathbb{Z}, \qquad j \in J \end{array} \tag{2.9}$$

Dengan $\mathbb{Z}$ adalah himpunan bilangan bulat, J adalah himpunan indeks, $J = \{1,2,\dots,n\}$

Linear Programming Relaxation (LP relaksasi) dari masalah pemrograman linier bilangan bulat adalah masalah pemrograman linier bilangan bulat dengan kendala $x_j \in \mathbb{Z}, j \in J$ dihilangkan. Sembarang masalah pemrograman linier bilangan bulat dapat dinyatakan sebagai LP relaksasi ditambah dengan suatu kendala, yaitu kendala yang menyatakan variabel bernilai bilangan bulat. Salah satu cara yang digunakan dalam menyelesaikan masalah pemrograman bilangan bulat adalah metode *branch and bound* yang akan dijelaskan di sub subbab berikut.

2.6.1 *Branch and Bound*

Salah satu cara yang sering digunakan untuk menyelesaikan masalah pemrograman linier bilangan bulat adalah metode *branch and bound*.

Konsep dasar teknik *branch and bound* adalah *divide and conquer*. Karena masalah awal terlalu besar untuk diselesaikan secara langsung, maka masalah awal akan dibagi (*divided*) menjadi submasalah-submasalah sampai submasalah-submasalah tersebut dapat diabaikan. Proses *dividing* dilakukan dengan membagi daerah *feasible* masalah awal menjadi daerah-daerah *feasible* yang lebih kecil. Sedangkan *conquering* (*fathoming*) dilakukan sebagian dengan membatasi (*bounding*) seberapa baik nilai optimal setiap submasalah dan mengabaikan submasalah yang tidak memuat solusi optimal masalah awal.

Berdasarkan penjelasan di atas, teknik *branch and bound* untuk masalah (2.9) memiliki tiga langkah dasar, yaitu :

1. *Branching* : Menetapkan nilai salah satu variabel, misalkan x_t menjadi 0 atau 1. Sehingga masalah utama terbagi menjadi 2 sub masalah, dimana submasalah pertama nilai dari $x_t = 1$, sedangkan submasalah kedua nilai dari $x_t = 0$.
2. *Bounding* : Untuk setiap submasalah, batas dari nilai solusi terbaik yang mungkin didapat dari solusi LP relaksasi submasalah tersebut. LP relaksasi dari masalah pemrograman bilangan bulat adalah masalah program linier dimana kendala tambahan (nilai dari variabel yang dicari harus berupa bilangan bulat) dihilangkan.
3. *Conquering/Fathoming* : Ada 3 cara untuk mengindikasi submasalah yang dianggap tidak perlu pengujian lebih lanjut, yaitu :
   1. Jika batas nilai solusi untuk submasalah tersebut lebih kecil atau sama dengan nilai solusi terbaik fungsi tujuan yang telah didapat.
   2. Jika LP relaksasi dari submasalah tidak mempunyai solusi *feasible.*
   3. Solusi optimal dari LP relaksasi, nilainya sudah berupa bilangan bulat.

2.7 *Column Generation*

Beberapa masalah program linier yang berkembang dari masalah kombinatorial menjadi sulit ditangani karena banyaknya variabel/kolom yang terlibat. Variabel ini biasanya merepresentasikan pola kemungkinan solusi. Kesulitan yang dihadapi adalah terlalu banyak kemungkinan pola yang memenuhi syarat kendala dari masalah program linier tersebut. Metode *column generation* merupakan salah satu metode yang cukup efisien untuk menyelesaikan masalah program linier, khususnya masalah pemrograman linier bilangan bulat, yang melibatkan banyak kolom yang sangat besar.

Metode *column generation* pertama kali digunakan oleh Gilmore dan Gomory dalam penyelesaian *cutting stock problem* satu dimensi (Lubbecke, M.E dan J. Desrosiers, 2002). Ide *column generation* adalah cukup dengan menggunakan subhimpunan dari himpunan kolom yang besar dalam menyelesaikan masalah, kemudian kolom baru ditambahkan ke dalam subhimpunan tersebut hanya saat diperlukan, yaitu ketika variabel yang bersesuaian dengan kolom tersebut berpotensi mengoptimalkan fungsi tujuan. Dalam *column generation* masalah didekomposisi menjadi 2 bagian, yaitu *master problem* dan subproblem.

Perhatikan masalah program linier berikut, dengan selanjutnya sebut sebagai *master problem* (MP) dan misalkan J adalah himpunan kolom dari MP, dengan $|J|$ berukuran sangat besar,

$$\begin{aligned} \text{minimum} \quad & f(x) = \sum_{j \in J} c_j\, x_j \\ \text{dk} \quad & \sum_{j \in J} a_{ij}\, x_j = b_i \\ & x_j \geq 0, j \in J \end{aligned}$$

Prosedur dalam *column generation* bersifat iteratif, dimana setiap iterasi terdiri dari :

- Optimisasi *Restricted Master Problem* (RMP) untuk menentukan nilai optimal terkini dari fungsi obyektif $f(\mathbf{x})$ dan pengali dual $\mathbf{y}$. RMP adalah *master problem* yang dibatasi pada subhimpunan dari himpunan variabel/kolom yang ada.
- Optimisasi subproblem, yaitu mencari, jika masih ada suatu kolom $j \in J$dengan *reduced cost* negatif ; $\mathbf{r_N} = c_j - \mathbf{y^T a_j} < 0$.

Langkah pertama dalam menyelesaikan MP atau inisialisasi metode *column generation* adalah dengan membentuk *restricted master problem* (RMP) yaitu MP yang hanya menggunakan subhimpunan J', dengan $J' \subset J$, dimana J adalah himpunan semua pola penjadwalan MK yang *feasible*. Salah satu cara dalam membentuk RMP adalah dengan memilih secara acak kolom yang mungkin mengoptimalkan fungsi objektif. Sehingga didapat RMP untuk masalah ini adalah:

$$\text{minimum} \qquad f(x) = \sum_{j \in J'} c_j\, x_j$$

$$\text{dk} \qquad \sum_{j \in J'} a_{ij}\, x_j = b_i$$

$$x_j > 0, J \subset J'$$

Algoritma *column generation* untuk menyelesaikan masalah di atas, adalah sebagai berikut:

1. Dekomposisi masalah menjadi 2 bagian, yaitu
    - Master Problem
    - Subproblem
2. Inisialisasi MP yaitu ambil $J' \subset J$, bentuk *Restricted Master Problem* (RMP), yaitu MP yang dibatasi pada J'.
3. Selesaikan RMP.

4. Didapat solusi optimal terkini dari RMP, $\bar{\mathbf{x}}$ dan variabel dual yang bersesuaian, $\bar{\mathbf{y}}$**.** Bentuk subproblem dengan menggunakan $\bar{\mathbf{y}}$.
5. Selesaikan subproblem. Jika *reduced cost* setiap variabel nonbasis, $\mathbf{r_N} = c_j - \mathbf{y^T a_j} \geq 0$, maka solusi optimal RMP juga merupakan solusi optimal untuk MP.
6. Jika tidak demikian, maka tambahkan kolom baru ke RMP.
7. Ulangi langkah dari 3 hingga didapat kondisi optimal.

2.10 *Branch and Price*

Algoritma *branch and price* memodifikasi strategi dasar *branch and bound* dengan usaha untuk menguatkan LP relaksasi dari *integer linier programming* dengan pertidaksamaan baru sebelum melakukan percabangan sebuah solusi pecahan. Prosedur *branch and price* berfokus pada *column generation*. LP relaksasi dari *master problem* yang diselesaikan dengan *column generation*, mungkin tidak memiliki solusi bilangan bulat dan menggunakan prosedur standar *branch and bound* yang ada tidak seperti halnya mencari solusi optimal, atau yang baik, atau mungkin *feasible* terhadap permasalahan sebenarnya (B. Cynthia, dkk.1996).

Branch and price merupakan gabungan dari *branch and bound* dengan metode *column generation* untuk menyelesaikan integer programming berskala besar. Di tiap *node* dari *tree branch and bound*, kolom mungkin dibangkitkan untuk menguatkan LP relaksasi. Dalam *branch and price*, kolom disisihkan dari LP relaksasi karena terdapat terlalu banyak kolom yang ditangani secara efisien. Kemudian untuk memeriksa keoptimalan dari sebuah solusi LP, subproblem yang disebut *pricing problem* yang merupakan bagian permasalahan untuk dual LP, diselesaikan untuk mencoba mengidentifikasi kolom yang akan masuk menjadi variabel basis. Jika terdapat kolom yang dimaksud tersebut, maka solusi LP dioptimasi kembali. Percabangan terjadi ketika tidak terdapat kolom yang dinilai dapat masuk menjadi variabel basis dan solusi LP tidak memenuhi batasan integral.

BAB 3

PENJADWALAN MATA KULIAH DENGAN PENDEKATAN *COLUMN GENERATION*

Pada bab ini akan dibahas tentang masalah penjadwalan mata kuliah (MK), metode penyelesaian dan juga contoh sederhananya. Terlebih dahulu akan dibahas masalah penjadwalan mata kuliah secara umum, beberapa asumsi dan variabel yang digunakan.

3.1 Masalah penjadwalan mata kuliah

Penjadwalan mata kuliah (MK) di universitas didefinisikan sebagai proses penempatan mata kuliah pada periode yang terdiri dari lima hari kerja dalam seminggu pada ruangan tertentu yang sesuai dengan jumlah mahasiswa yang terdaftar di tiap mata kuliah (S. Daskalaki, et al., 2003). Seperti permasalahan lainnya dalam optimisasi kombinatorik, penjadwalan MK mempunyai beberapa teknik pendekatan yang diketahui dalam riset operasi dan ilmu komputer (S. Daskalaki, dkk., 2003).

3.1.1 Aturan untuk penjadwalan mata kuliah

Untuk membuat penjadwalan MK diperlukan aturan-aturan yang harus terpenuhi sebagai berikut:

1. MK yang dijadwalkan tidak boleh ada yang bentrok. Dua atau lebih MK dikatakan bentrok jika dijadwalkan di waktu yang sama untuk MK yang diajarkan dosen yang sama, diikuti kelompok mahasiswa yang sama, atau ditempatkan pada ruangan yang sama.
2. Penjadwalan harus lengkap. Penjadwalan dikatakan lengkap ketika semua MK yang direncanakan untuk setiap kelompok mahasiswa dalam satu semester terdapat di jadwal, dengan jumlah waktu yang tepat untuk tiap MK.

3. Pemilihan waktu khusus oleh dosen dapat dituruti jika memungkinkan. Tiap dosen mungkin mempunyai keinginan mengenai waktu pengajaran untuk MK yang diampunya. Sebagai contoh, ada dosen yang ingin mengajar hanya di pagi hari karena di siang hari atau sore hari ada kegiatan di luar mengajar.

Dari aturan-aturan di atas akan didefinisikan parameter-parameter terkait dengan penjadwalan MK:

- Slot-waktu: dalam 1 hari didefinisikan 4 slot-waktu dimana masing-masing slot-waktu terdiri dari 2 jam. Periode waktu yang digunakan adalah dari jam 8 pagi sampai jam 5 sore. Himpunan semua slot-waktu dalam 1 minggu dinotasikan dengan H, $H = \{1, 2, 3, \dots, 20\}$.
- Kelompok mahasiswa: Dalam hal ini yang digunakan adalah kelompok MK per-semester. Misalkan semester genap kelompoknya adalah MK untuk semester 2, semester 4, dan semester 6. Himpunan semua kelompok mahasiswa dinotasikan dengan Q, $Q = \{\text{grup\#1}, \text{grup\#2}, \dots, \text{grup\#}|Q|\}$.
- Mata kuliah yang dijadwalkan. Himpunan semua MK yang dijadwalkan dinotasikan dengan C, $C = \{\text{MK\#1}, \text{MK\#2}, \dots, \text{MK\#}|C|\}$.
- Ruangan yang tersedia. Ruangan terbagi menjadi beberapa jenis dimana setiap MK akan ditempatkan di jenis ruangan yang cocok.
  Himpuan semua ruangan yang tersedia dinotasikan dengan,
  $K = \{\text{ruangan\#1}, \text{ruangan\#2}, \dots, \text{ruangan\#}|K|\}$.

Definisikan dua parameter, yaitu $a_{(kh)(jc)}$, $a'_{(h)(jc)}$, dan variabel keputusan x_{jc} dengan $k \in K, h \in H, c \in C, j \in P(c)$. Himpunan $P(c)$ adalah himpunan pola *feasible* untuk MK c. Parameter $a_{(kh)(jc)}$ bernilai 1, ketika MK c ditempatkan di ruangan k, slot-waktu h untuk pola j dan bernilai 0 untuk lainnya. Parameter $a'_{(h)(jc)}$ bernilai 1, ketika MK c ditempatkan di slot-waktu h untuk pola j dan bernilai 0 untuk lainnya. Sehingga jelas bahwa nilai $a'_{(h)(jc)} = 0$ jika dan hanya jika nilai $a_{(kh)(jc)} = 0$ untuk setiap $k \in K$. Variabel keputusan x_{jc} bernilai 1, ketika MK c menggunakan pola j dan bernilai 0 untuk lainnya. .

Setiap MK mempunyai pola penjadwalan yang berbeda-beda dimana nantinya mata kuliah-mata kuliah tersebut akan dijadwalkan di slot-waktu dan ruangan tertentu, oleh karena itu ada beberapa asumsi yang akan digunakan dalam masalah penjadwalan, yaitu:

1. Ruangan dapat dibagi menjadi himpunan ruangan dengan jenis yang sama
2. Jadwal yang dibuat adalah jadwal mingguan
3. Dalam satu minggu dibagi menjadi beberapa slot-waktu, dengan pembagian yang sama (satu atau dua jam per slot-waktu)

Dengan asumsi-asumsi di atas, maka akan didefinisikan variabel-variabel yang akan digunakan untuk penjadwalan MK berikut ini:

Variabel yang digunakan terkait dengan ruangan adalah :

- K = Himpunan jenis ruangan
- n_k= Banyaknya ruangan jenis k dengan $k \in K$

Untuk MK dan hubungan antar ruangan dengan MK, variabel yang digunakan adalah:

- C = Himpunan MK
- $K(c)$ = Himpunan jenis ruangan yang *feasible* untuk mata kuliah c, dengan $c \in C$

Dan juga untuk slot-waktu dan hubungannya dengan MK, variabel yang digunakan adalah:

- H= Himpunan slot-waktu
- $d(c)$ = Banyaknya slot-waktu yang disyaratkan untuk mata kuliah c

Adakalanya terdapat beberapa MK yang diajarkan oleh dosen yang sama dan beberapa MK yang diberikan untuk kelompok mahasiswa tertentu. Maka mata kuliah-mata kuliah tersebut dikelompokkan menjadi satu kelompok. Untuk pendefinisian kelompok dibutuhkan variabel-variabel berikut:

- C_q= Himpunan MK yang tidak boleh bentrok, dengan $C_q \subset C$ dan $q \in Q$ (Q adalah himpunan indeks)

Periode yang ingin dijadwalkan adalah 1 minggu, maka dalam 1 minggu terdapat pola tiap MK yang ditempatkan di slot-waktu dan ruangan tertentu. Untuk pola tiap mata kuliah didefinisikan variabel berikut:

- $P(c)$ = Himpunan pola penjadwalan untuk MK c, dengan $c \in C$

Setelah pendefinisian variabel dan asumsi-asumsi di atas selanjutnya akan dibahas pemodelan matematika dari penjadwalan MK.

3.2 Model Matematika

Jumlah pola untuk tiap MK adalah eksponensial (Q. Andrea dan S. Paolo), tetapi dalam pengerjaannya nantinya akan dipilih subset dari pola tiap MK. Untuk membatasi pola-pola tersebut definisikan:

$$a_{(kh)(jc)} = \begin{cases} 1, & \text{jika mata kuliah } c \text{ ditempatkan di ruangan} \\ & \text{jenis } k \text{ slot} - \text{waktu } h \text{ untuk pola } j \epsilon P(c) \\ 0, & \text{lainnya} \end{cases} \qquad (3.1.1)$$

$$a'_{(h)(jc)} = \begin{cases} 1, & \text{jika mata kuliah } c \text{ ditempatkan di} \\ & \text{slot} - \text{waktu } h \text{ untuk pola } j \epsilon P(c) \\ 0, & \text{lainnya} \end{cases} \qquad (3.1.2)$$

$$x_{jc} = \begin{cases} 1, & \text{jika pola } j \epsilon P(c) \text{ digunakan untuk mata kuliah } c \\ 0, & \text{lainnya} \end{cases} \qquad (3.1.3)$$

Tiap dosen mungkin ingin menyatakan pendapat tentang pemilihan slot-waktu untuk MK yang diampunya. Oleh karena itu dosen dibolehkan untuk memilih sendiri slot-waktu untuk MK yang diampunya, maka akan didefinisikan parameter terkait pemilihan tersebut:

- s_{khc} = pemilihan oleh dosen menggunakan ruangan jenis k dan slot-waktu h untuk MK $c \in C$

Nilai dari parameter s_{khc} adalah 0, 1, atau 2 untuk suatu slot-waktu tertentu. Bernilai 0 jika dosen menyatakan tidak berkeinginan untuk mengajar di slot-waktu tersebut. Bernilai 2 jika dosen menyatakan sangat berkeinginan untuk mengajar di slot-waktu tersebut, dan bernilai 1 jika dosen menyatakan bisa untuk mengajar di slot-waktu tersebut.

Memaksimalkan nilai pemilihan slot-waktu oleh dosen inilah yang akan dijadikan fungsi tujuan yang akan dicapai.

Berdasarkan pendefinisian variabel dan parameter di atas, maka masalah penjadwalan MK dapat diformulasikan sebagai *integer linear programming* (ILP) *problem* sebagai berikut:

- Memaksimumkan pemilihan slot-waktu oleh dosen, berarti

$$\text{maksimum} \sum_{c\epsilon C} \sum_{j\epsilon P\,(c)} \left(\sum_{h\epsilon H} \sum_{k\epsilon K} a_{(kh)(jc)}\, s_{skc} \right) x_{jc}$$

- Dengan kendala sebagai berikut
  - Ruangan yang cocok untuk MK c ditempatkan tidak melebihi banyaknya ruangan jenis k, berarti

$$\sum_{c\epsilon C} \sum_{j\epsilon P\,(c)} a_{(kh)(jc)} x_{jc} \leq n_k \; k\epsilon K, \quad h\epsilon H$$

  - MK c dalam satu kelompok hanya dipilih minimal satu pola MK, berarti

$$\sum_{c\epsilon\, C_q} \sum_{j\epsilon P\,(c)} a'_{(h)(jc)} x_{jc} \leq 1 \qquad h\epsilon H, \quad q\epsilon Q$$

  - Untuk setiap MK hanya satu pola penjadwalan yang digunakan, berarti

$$\sum_{j\epsilon P\,(c)} x_{jc} = 1 \qquad c\epsilon C$$

Sehingga model matematis untuk masalah penjadwalan kuliah adalah sebagai berikut:

$$\text{maksimum} \sum_{c\epsilon C} \sum_{j\epsilon P\,(c)} \left(\sum_{h\epsilon H} \sum_{k\epsilon K} a_{(kh)(jc)}\, s_{skc} \right) x_{jc} \tag{3.1}$$

$$\text{dk} \qquad \sum_{c\epsilon C} \sum_{j\epsilon P\,(c)} a_{(kh)(jc)} x_{jc} \leq n_k \; k\epsilon K, \quad h\epsilon H \tag{3.2}$$

$$\sum_{c\epsilon\, C_q} \sum_{j\epsilon P\,(c)} a'_{(h)(jc)} x_{jc} \leq 1 \qquad h\epsilon H, \quad q\epsilon Q \tag{3.3}$$

$$\sum_{j \epsilon P\,(c)} x_{jc} = 1 \qquad c \epsilon C \tag{3.4}$$

$$x_{jc} \in \{0,1\} \tag{3.5}$$

.

Masalah ILP di atas akan diselesaikan dengan pendekatan *column generation*. Dimana sebelumnya akan dibahas metode *branch and price* yang merupakan gabungan antara metode *column generation* dan *branch and bound.*

3.3 Branch and Price

Branch and price merupakan gabungan dari *branch and bound* dengan metode *column generation* untuk menyelesaikan integer programming berskala besar. Algoritma *branch and price* memodifikasi strategi dasar *branch and bound* dengan usaha untuk menguatkan program linear relaksasi (LP relaksasi) dari pemrograman bilangan bulat. Di tiap *node* dari *tree branch and bound* kolom mungkin dibangkitkan untuk menguatkan LP relaksasi. Prosedur *branch and price* berfokus pada *column generation*.

3.3.1 Column Generation

Masalah penjadwalan MK melibatkan sejumlah ruangan dan slot-waktu, dimana setiap MK mempunyai pola penjadwalan yang sangat beragam, sehingga terdapat sejumlah besar kombinasi yang mungkin untuk membuat suatu jadwal, dengan perkataan lain ILP disini melibatkan jumlah kolom yang besar, sehingga diperlukan metode *column generation* untuk menyelesaikan masalah tersebut.

Langkah pertama metode *column generation* adalah membagi masalah menjadi dua bagian, yaitu:

1. Masalah ILP sebagai *master problem*
2. Subproblem (*pricing problem*)

Dalam penyelesaiannya masalah ILP memilih suatu jadwal yang mungkin dan *feasible* yang telah diketahui untuk setiap MK, sedangkan subproblem digunakan untuk mendapatkan suatu jadwal yang baru untuk memperbaiki solusi dari masalah ILP.

3.3.2 Masalah *Integer Linear Programming* (ILP)

Langkah pertama dalam menyelesaikan masalah ILP pada persamaan (3.1)-(3.5) atau inisialisasi metode *column generation* adalah dengan membentuk *restricted master problem* (RMP) yaitu masalah ILP yang hanya menggunakan subhimpunan J', dengan $J' \subset J$, dimana J adalah himpunan semua jadwal MK yang layak. Salah satu cara dalam membentuk RMP adalah dengan memilih secara acak kolom yang mungkin mengoptimalkan fungsi tujuan. Sehingga didapat RMP untuk masalah ini adalah:

$$\text{maksimum} \sum_{c \epsilon C} \sum_{j \epsilon P\,(c)} \left(\sum_{h \epsilon H} \sum_{k \epsilon K} a_{(kh)(jc)}\, s_{skc} \right) x_{jc} \tag{3.6}$$

$$\text{dk} \quad \sum_{c \epsilon C} \sum_{j \epsilon P\,(c)} a_{(kh)(jc)} x_{jc} \leq n_k \quad k \epsilon K, \quad h \epsilon H \tag{3.7}$$

$$\sum_{c \epsilon C_q} \sum_{j \epsilon P\,(c)} a'_{(h)(jc)} x_{jc} \leq 1 \qquad h \epsilon H, \quad q \epsilon Q \tag{3.8}$$

$$\sum_{j \epsilon P\,(c)} x_{jc} = 1 \qquad c \epsilon C \tag{3.9}$$

$$x_{jc} \epsilon \{0,1\} \qquad j \epsilon J' \tag{3.10}$$

Selanjutnya, bentuk LP relaksasi dari bentuk masalah ILP di atas, yaitu dengan menghilangkan kendala (3.10). Kemudian LP relaksasi diselesaikan dengan metode simpleks.

Berikutnya pada saat keadaan optimal dicapai di RMP yaitu di himpunan J', akan diperiksa apakah keadaan optimal tersebut berlaku juga untuk *master problem,* yaitu dengan cara meyelesaikan subproblem yang akan dijelaskan berikut:

3.3.3 Subproblem (*Pricing Problem*)

Misalkan didefinisikan variabel dual w_{kh}, v_{hq}, dan u_c untuk kendala (3.7), (3.8), dan (3.9) secara berurutan, sehingga permasalahan dual adalah:

$$\sum_{h\epsilon H}\sum_{k\epsilon K} a_{(kh)(jc)}\, w_{kh} + \sum_{h\epsilon H}\sum_{q\epsilon Q\,(c)} a'_{(h)(jc)}\, v_{hq} + u_c \geq \sum_{h\epsilon H}\sum_{k\epsilon K} a_{(kh)(jc)}\, s_{khc}\; j\epsilon P(c), c\epsilon C$$

atau

$$\sum_{h\epsilon H}\left(\sum_{k\epsilon K} a_{(kh)(jc)}(w_{kh} - s_{khc}) + \sum_{q\epsilon Q} a'_{(h)(jc)}\, v_{hq}\right) + u_c \geq 0 \qquad j\epsilon P(c), c\epsilon C \qquad (3.11)$$

Untuk memilih variabel masuk dari luar J', yaitu pola penjadwalan yang tidak terdapat di J', akan diminimumkan persamaan (3.11) terhadap a dan a'.

$$\sum_{h\epsilon H}\left(\sum_{k\epsilon K} a_{(kh)(jc)}(w_{kh} - s_{khc}) + \sum_{q\epsilon Q} a'_{(h)(jc)}\, v_{hq}\right) \qquad (3.12)$$

Sehingga meminimumkan (3.12) sama dengan meminimumkan $(w_{kh} - s_{khc})$ dan $\sum_{q\in Q} v_{hq}$, misalkan didefinisikan

$$\acute{w}_{hc} = \min_{k\epsilon K\,(c)} w_{kh} - s_{khc}$$

$$\acute{v}_{hc} = \sum_{q\epsilon Q\,(c)} v_{hq}$$

dan

$$t_{hc} = \acute{w}_{hc} + \acute{v}_{hc}$$

Sehingga meminimumkan (3.12) ekivalen dengan meminimumkan:

$$\sum_{h\epsilon H} t_{hc}\, a'_{(h)(jc)} \qquad (3.13)$$

Misalkan nilai minimum (3.13) adalah M_c, jika $M_c + u_c < 0$ maka kondisi optimal untuk J' belum optimal untuk *master problem* untuk MK c, sehingga pola yang didapatkan dari meminimumkan (3.13) dimasukan ke RMP . Sebaliknya jika $M_c + u_c \geq 0$ maka kondisi optimal untuk J' sudah optimal untuk *master problem* untuk MK c.

3.3.4 *Branch and Bound*

Ketika LP relaksasi dari RMP dioptimisasi, metode *branch and bound* digunakan saat solusi yang didapat bukan bilangan bulat. Di tiap *node* dari *tree branch and bound* dilakukan proses *column generation* dimulai dari *node* awal, untuk memberikan batas valid yang digunakan untuk *fathoming*. Perlu diperhatikan, ketika memproses suatu node tidak dibutuhkan untuk menyelesaikan LP sampai optimal. Alasan utama untuk menyelesaikan LP sampai optimal adalah adanya kemungkinan untuk memotong *node* berdasarkan batas (*bound*). Pemangkasan masih mungkin terjadi tanpa menyelesaikan LP sampai optimal jika dapat menghitung batas yang tepat.

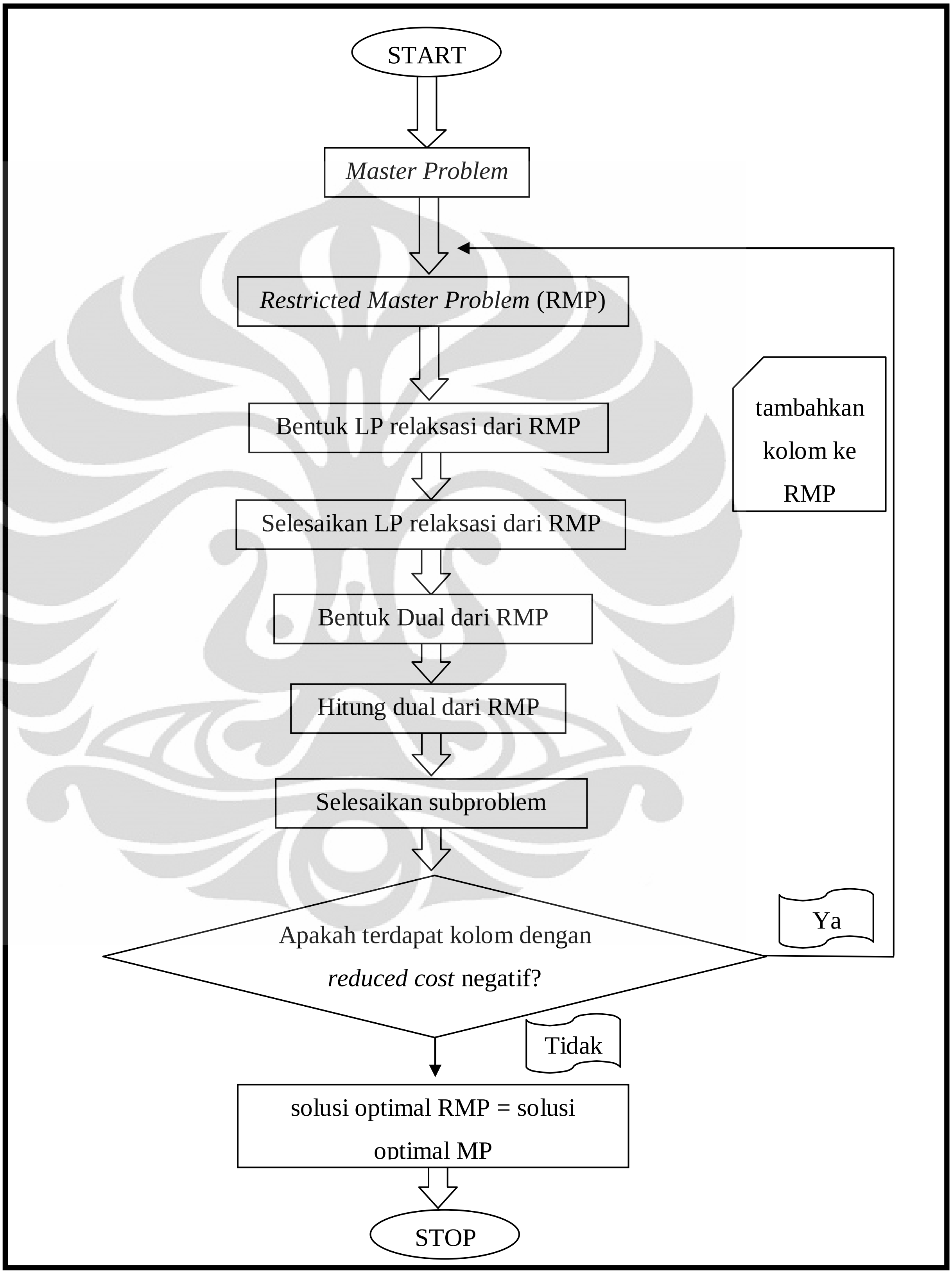
Misalkan $\hat{x}_{jk}$ untuk setiap j dan k adalah solusi yang didapatkan dari metode *column generation* di suatu *node* di *tree branch and bound*, sehingga aturan mempartisi atau percabangan RMP menjadi 2 subproblem yaitu:

1. $x_{jk} \leq \lfloor \hat{x}_{jk} \rfloor$
2. $x_{jk} \geq \lceil \hat{x}_{jk} \rceil$

dimana $\lfloor x \rfloor$ adalah bilangan bulat terbesar yang lebih kecil atau sama dengan x dan $\lceil x \rceil$ adalah bilangan bulat terkecil yang lebih besar atau sama dengan x.

3.3.5 *Flowchart* metode *column generation*

Agar lebih jelas mengenai metode *column generation*, dapat dilihat pada *flowchart* berikut ini:



Gambar 3.1: *Flowchart* metode *column generation*

Berikut ini akan diberikan contoh sederhana masalah penjadwalan 5 MK dengan pendekatan *column generation*.

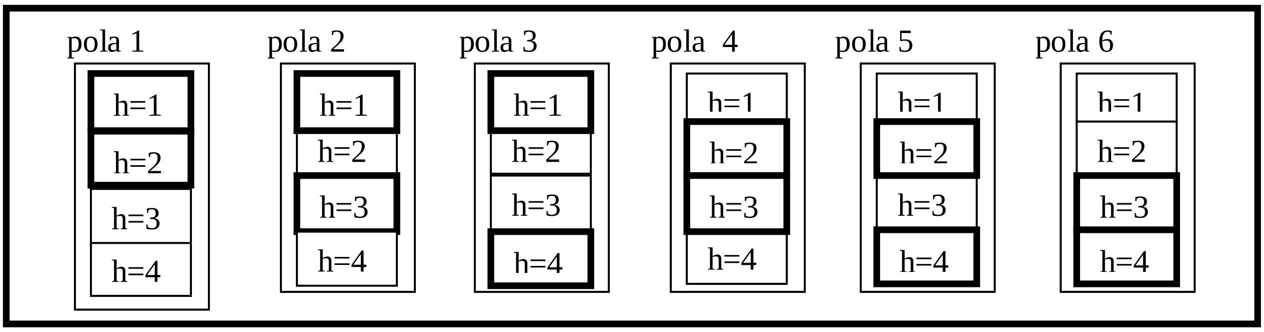
Misalkan terdapat 4 slot-waktu (H=4) dan 5 mata kuliah yaitu c_1, c_2, c_3, c_4, c_5 dengan ketentuan $d(c_1) = 2$, $d(c_2) = 2$, $d(c_3) = 1$, $d(c_4) = 1$, $d(c_5) = 2$. Tiap MK dikelompokkan ke dalam 2 kelompok , yaitu $C_1 = \{c_1, c_2\}$ dan $C_2 = \{c_3, c_4, c_5\}$, dimana kelompok tersebut adalah kelompok MK yang tidak boleh saling bentrok. Kemudian terdapat 2 jenis kelas ($K = \{k_1, k_2\}$), dengan tiap jenis ruangan masing-masing sebanyak dua atau $n(k_1) = n(k_2) = 2$. Hubungan antara MK dan jenis kelas adalah $K(c_1) = \{k_1\}$, $K(c_2) = K(c_5) = \{k_2\}, K(c_3) = K(c_4) = \{k_1, k_2\}$. Karena ruangan yang tepat untuk MK c_3 dan c_4 adalah k_1 dan k_2, asumsikan bahwa ruangan yang tepat untuk MK c_3 dan c_4 adalah k_1. Misalkan pemilihan dosen untuk tiap MK adalah:

Tabel 3.1:. Nilai pemilihan slot-waktu oleh dosen setiap MK

	c_1	c_2	c_3	c_4	c_5
h=1	0	2	1	2	2
h=2	2	0	1	0	2
h=3	1	1	0	1	1
h=4	2	1	2	1	0

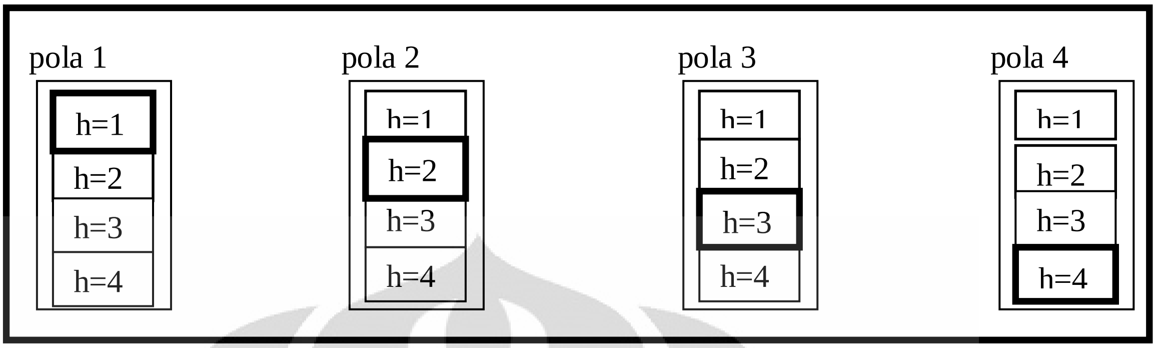
Sebagai ilustrasi, akan digambarkan beberapa pola penjadwalan MK:

Untuk MK yang membutuhkan 2 slot-waktu, polanya adalah: $C_2^4 = 6$



Gambar 3.2: Pola penjadwalan untuk MK yang membutuhkan 2 slot waktu

Untuk MK yang mensyaratkan 1 slot-waktu, polanya adalah: $C_1^4 = 4$



Gambar 3.3: Pola penjadwalan untuk MK yang membutuhkan 1 slot waktu

Dari 6 pola penjadwalan di atas, maka dapat diperoleh model matematikanya sebagai berikut:

- Fungsi objektif

$$\text{maksimum} \quad \sum_{c\in C} \sum_{j\in P(c)} \Big(\sum_{h\in H} \sum_{k\in K} a_{(kh)(jc)} s_{khc}\Big) x_{jc}$$

Jika dijabarkan menjadi

$$\text{maksimum} \quad \sum_{c\in C} \sum_{j\in P(c)} \big(a_{(k_1 1)(jc)} s_{k_1 1c} + a_{(k_2 1)(jc)} s_{k_2 1c} + \cdots$$
$$+ a_{(k_1 4)(jc)} s_{k_1 4c} + a_{(k_2 4)(jc)} s_{k_2 4c}\big) x_{jc}$$

atau

$$\text{maksimum} \quad \Big(\big(a_{(k_1 1)(1a)} s_{k_1 1a} + a_{(k_2 1)(1a)} s_{k_2 1a} + \cdots + a_{(k_1 4)(1a)} s_{k_1 4a}$$
$$+ a_{(k_2 4)(1a)} s_{k_2 4a}\big) x_{1a} + \cdots$$
$$+ \big(a_{(k_1 1)(6a)} s_{k_1 1a} + a_{(k_2 1)(6a)} s_{k_2 1a} + \cdots + a_{(k_1 4)(6a)} s_{k_1 4a}$$
$$+ a_{(k_2 4)(6a)} s_{k_2 4a}\big) x_{6a}\Big) + \cdots$$
$$+ \Big(\big(a_{(k_1 1)(1e)} s_{k_1 1e} + a_{(k_2 1)(1e)} s_{k_2 1a} + \cdots + a_{(k_1 4)(1e)} s_{k_1 4e}$$
$$+ a_{(k_2 4)(1e)} s_{k_2 4e}\big) x_{1e} + \cdots$$
$$+ \big(a_{(k_1 1)(6e)} s_{k_1 1e} + a_{(k_2 1)(6e)} s_{k_2 1e} + \cdots + a_{(k_1 4)(6e)} s_{k_1 4e}$$
$$+ a_{(k_2 4)(6e)} s_{k_2 4e}\big) x_{6e}\Big)$$

Menerapkan aturan (3.1.1) dengan nilai pemilihan tiap MK seperti Tabel 3.1 di atas, maka didapatkan perhitungan sebagai berikut:

$$\text{maksimum} \quad (2x_{1a}+x_{2a}+2x_{3a}+3x_{4a}+4x_{5a}+3x_{6a}) + (2x_{1b}+3x_{2b}+3x_{3b}+x_{4b}+x_{5b}+2x_{6b}) + (x_{1c}+x_{2c}+2x_{4c}) + (2x_{1d}+x_{3d}+x_{4d}) + (4x_{1e}+3x_{2e}+2x_{3e}+3x_{4e}+2x_{5e}+x_{6e})$$

- Kendala (3.2)

$$\sum_{c\epsilon C}\sum_{j\epsilon P\,(c)} a_{(kh)(jc)}x_{jc} \le n_k \text{ untuk } k = k_1, k_2\,, \qquad h = 1,2,3,4$$

Yaitu:

$$\left(a_{(kh)(1a)}x_{1a}+\cdots+a_{(kh)(6a)}x_{6a}\right)+\left(a_{(kh)(1b)}x_{1b}+\cdots+a_{(kh)(6a)}x_{6b}\right) + \left(a_{(kh)(1c)}x_{1c}+\cdots+a_{(kh)(4c)}x_{4c}\right) + \left(a_{(kh)(1d)}x_{1d}+\cdots+a_{(kh)(4d)}\right) + (a_{(kh)(1e)}x_{1e}+\cdots+a_{(kh)(6e)}x_{6e}) \le n_k$$

Dengan menerapkan aturan (3.1.1), didapatkan:

- Untuk $k = k_1$ dan h=1, nilai parameter $a_{(kh)(1a)} = a_{(kh)(2a)} = a_{(kh)(3a)} = a_{(kh)(1c)} = a_{(kh)(1d)} = 1$ dan lainnya bernilai 0, sehingga diperoleh

$x_{1a}+x_{2a}+x_{3a}+0+0+0)+(0)+(x_{1c}+0+0+0)+(x_{1d}+0+0+0)$ $+(0) \le 2$

atau

$$x_{1a}+x_{2a}+x_{3a}+x_{1c}+x_{2d} \le 2$$

Dengan cara yang sama seperti di atas diperoleh kendala-kendala lainnya.

- Untuk $k = k_1$dan h=2

$$x_{1a}+x_{4a}+x_{5a}+x_{2c}+x_{2d} \le 2$$

- Untuk $k = k_1$dan h=3

$$x_{2a}+x_{4a}+x_{6a}+x_{3c}+x_{3d} \le 2$$

- Untuk $k = k_1$dan h=4

$$x_{3a}+x_{5a}+x_{6a}+x_{4c}+x_{4d} \le 2$$

- Untuk $k = k_2$dan h=1

$$x_{1b}+x_{2b}+x_{3b}+x_{1e}+x_{2e}+x_{3e} \le 2$$

- Untuk $k = k_2$dan h=2

$$x_{1b} + x_{4b} + x_{5b} + x_{1e} + x_{4e} + x_{5e} \leq 2$$

- Untuk $k = k_2$dan h=3

$$x_{2b} + x_{4b} + x_{6b} + x_{2e} + x_{4e} + x_{6e} \leq 2$$

- Untuk $k = k_2$dan h=4

$$x_{3b} + x_{5b} + x_{6b} + x_{3e} + x_{5e} + x_{6e} \leq 2$$

- Kendala (3.3)

$$\sum_{c \epsilon C} \sum_{j \epsilon P\ (c)} a'_{(h)(jc)} x_{jc} \leq 1 \quad \text{untuk}\ h = 1, 2, 3, 4 \text{ dan } q = 1, 2$$

Yaitu:

$$\begin{aligned} &\left(\left(a'_{(h)(1a)} x_{1a} + \cdots + a'_{(h)(6a)} x_{6a}\right)\right. \\ &\quad \left. + \left(a'_{(h)(1b)} x_{1b} + \cdots + a'_{(h)(6b)} x_{6b}\right)\right) \\ &\quad + \left(\left(a'_{(h)(1c)} x_{1c} + \cdots + a'_{(h)(4c)} x_{4c}\right)\right. \\ &\quad + \left(a'_{(h)(1d)} x_{1d} + \cdots + a'_{(h)(4d)} x_{4d}\right) \\ &\quad \left. + \left(a'_{(h)(1e)} x_{1e} + \cdots + a'_{(h)(6e)} x_{1e}\right)\right) \end{aligned}$$

- Untuk q = 1 dan h = 1

$$x_{1a} + x_{2a} + x_{3a} + x_{1b} + x_{2b} + x_{3b} \leq 1$$

- Untuk q = 1 dan h = 2

$$x_{1a} + x_{4a} + x_{5a} + x_{1b} + x_{4b} + x_{5b} \leq 1$$

- Untuk q = 1 dan h = 3

$$x_{2a} + x_{4a} + x_{6a} + x_{2b} + x_{4b} + x_{6b} \leq 1$$

- Untuk q = 1 dan h = 4

$$x_{3a} + x_{5a} + x_{6a} + x_{3b} + x_{5b} + x_{6b} \leq 1$$

- Untuk q = 2 dan h = 1

$$x_{1c} + x_{1d} + x_{1e} + x_{2e} + x_{3e} \leq 1$$

- Untuk q = 2 dan h = 2

$$x_{2c} + x_{2d} + x_{1e} + x_{4e} + x_{5e} \leq 1$$

- Untuk q = 2 dan h = 3

$$x_{3c} + x_{3d} + x_{2e} + x_{4e} + x_{6e} \leq 1$$

- Untuk q = 2 dan h = 4

$$x_{4c} + x_{4d} + x_{3e} + x_{5e} + x_{6e} \leq 1$$

- Kendala (3.4)

$$\sum_{j \epsilon P\ (c)} x_{jc} = 1\ , \quad c = c_1, c_2, c_3, c_4, c_5$$

- Untuk MK $c_1 = a$

$$x_{1a} + x_{2a} + x_{3a} + x_{4a} + x_{5a} + x_{6a} = 1$$

- Untuk MK $c_2 = b$

$$x_{1b} + x_{2b} + x_{3b} + x_{4b} + x_{5b} + x_{6b} = 1$$

- Untuk MK $c_3 = c$

$$x_{1c} + x_{2c} + x_{3c} + x_{4c} = 1$$

- Untuk MK $c_4 = d$

$$x_{1d} + x_{2d} + x_{3d} + x_{4d} = 1$$

- Untuk MK $c_5 = e$

$$x_{1e} + x_{2e} + x_{3e} + x_{4e} + x_{5e} + x_{6e} = 1$$

Sehingga *master problem* dari permasalahan penjadwalan 5 MK di atas adalah sebagai berikut:

$$\begin{aligned} \text{maksimum} \quad & (2x_{1a} + x_{2a} + 2x_{3a} + 3x_{4a} + 4x_{5a} + 3x_{6a}) \\ & + (2x_{1b} + 3x_{2b} + 3x_{3b} + x_{4b} + x_{5b} + 2x_{6b}) \\ & + (x_{1c} + x_{2c} + 2x_{4c}) + (2x_{1d} + x_{3d} + x_{4d}) + \\ & (4x_{1e} + 3x_{2e} + 2x_{3e} + 3x_{4e} + 2x_{5e} + x_{6e}) \end{aligned}$$

dk

$$x_{1a} + x_{2a} + x_{3a} + x_{1c} + x_{1d} \leq 2 \quad (3.2.1)$$

$$x_{1a} + x_{4a} + x_{5a} + x_{2c} + x_{2d} \leq 2 \quad (3.2.2)$$

$$x_{2a} + x_{4a} + x_{6a} + x_{3c} + x_{3d} \leq 2 \quad (3.2.3)$$

$$x_{3a} + x_{5a} + x_{6a} + x_{4c} + x_{4d} \leq 2 \quad (3.2.4)$$

$$x_{1b} + x_{2b} + x_{3b} + x_{1e} + x_{2e} + x_{3e} \leq 2 \quad (3.2.5)$$

$$x_{1b} + x_{4b} + x_{5b} + x_{1e} + x_{4e} + x_{5e} \leq 2 \quad (3.2.6)$$

$$x_{2b} + x_{4b} + x_{6b} + x_{2e} + x_{4e} + x_{6e} \leq 2 \quad (3.2.7)$$

$$x_{3b} + x_{5b} + x_{6b} + x_{3e} + x_{5e} + x_{6e} \leq 2 \quad (3.2.8)$$

$$x_{1a} + x_{2a} + x_{3a} + x_{1b} + x_{2b} + x_{3b} \leq 1 \quad (3.3.1)$$

$$x_{1a} + x_{4a} + x_{5a} + x_{1b} + x_{4b} + x_{5b} \leq 1 \quad (3.3.2)$$

$$x_{2a} + x_{4a} + x_{6a} + x_{2b} + x_{4b} + x_{6b} \leq 1 \quad (3.3.3)$$
$$x_{3a} + x_{5a} + x_{6a} + x_{3b} + x_{5b} + x_{6b} \leq 1 \quad (3.3.4)$$
$$x_{1c} + x_{1d} + x_{1e} + x_{2e} + x_{3e} \leq 1 \quad (3.3.5)$$
$$x_{2c} + x_{2d} + x_{1e} + x_{4e} + x_{5e} \leq 1 \quad (3.3.6)$$
$$x_{3c} + x_{3d} + x_{2e} + x_{4e} + x_{6e} \leq 1 \quad (3.3.7)$$
$$x_{4c} + x_{4d} + x_{3e} + x_{5e} + x_{6e} \leq 1 \quad (3.3.8)$$
$$x_{1a} + x_{2a} + x_{3a} + x_{4a} + x_{5a} + x_{6a} = 1 \quad (3.4.1)$$
$$x_{1b} + x_{2b} + x_{3b} + x_{4b} + x_{5b} + x_{6b} = 1 \quad (3.4.2)$$
$$x_{1c} + x_{2c} + x_{3c} + x_{4c} = 1 \quad (3.4.3)$$
$$x_{1d} + x_{2d} + x_{3d} + x_{4d} = 1 \quad (3.4.4)$$
$$x_{1e} + x_{2e} + x_{3e} + x_{4e} + x_{5e} + x_{6e} = 1 \quad (3.4.5)$$
$$x_{1a}, x_{2a}, x_{3a}, x_{4a}, x_{5a}, x_{6a}, x_{1b}, x_{2b}, x_{3b}, x_{4b}, x_{5b}, x_{6b}, x_{1c},$$
$$x_{2c}, x_{4c}, x_{1d}, x_{2d}, x_{3d}, x_{4d}, x_{1e}, x_{2e}, x_{3e}, x_{4e}, x_{5e}, x_{6e} \in \{0, 1\} \quad (3.5)$$

Setelah membentuk model matematika penjadwalan MK, selanjutnya *master problem* akan diselesaikan dengan pendekatan *column generation*.

Iterasi Pertama (RMP1)

Inisialisasi dalam menyelesaikan *master problem* adalah dengan membentuk *restricted master problem* (RMP), yaitu *master problem* yang hanya menggunakan pola pada J', dengan $J' \subset J$, dimana J adalah himpunan semua pola penjadwalan yang *feasible*. Salah satu caranya adalah memilih secara acak kolom yang mungkin mengoptimalkan fungsi tujuan.

Misalkan pilih tiga pola secara acak dari tiap mata kuliah. Misalkan untuk MK c_1, pola yang terpilih adalah pola ke-2 , ke-5, dan ke-4, untuk MK c_2 adalah pola ke-5, ke-3, dan ke-4, untuk MK c_3 adalah pola ke-1, ke-4, ke-3, untuk MK c_4 adalah pola ke-2, ke-1, dan ke-3, untuk MK c_5 adalah pola ke-6, ke-1, dan ke-4.

Sehingga didapatkan untuk RMP1 dengan membentuk LP relaksasi yaitu menghilangkan kendala (3.5) sebagai berikut:

$$\text{maksimum} \quad (x_{2a} + 3x_{4a} + 4x_{5a}) + (3x_{3b} + x_{4b} + x_{5b})$$
$$+ (x_{1c} + 0x_{3c} + 2x_{4c}) + (2x_{1d} + 0x_{2d} + x_{3d}) +$$

$$(4x_{1e} + 3x_{4e} + x_{6e})$$

dk

$$x_{2a} + x_{1c} + x_{1d} \leq 2$$
$$x_{4a} + x_{5a} + x_{2d} \leq 2$$
$$x_{2a} + x_{4a} + x_{3c} + x_{3d} \leq 2$$
$$x_{5a} + x_{4c} \leq 2$$
$$x_{3b} + x_{1e} \leq 2$$
$$x_{4b} + x_{5b} + x_{1e} + x_{4e} \leq 2$$
$$x_{4b} + x_{4e} + x_{6e} \leq 2$$
$$x_{3b} + x_{5b} + x_{6e} \leq 2$$
$$x_{2a} + x_{3b} \leq 1$$
$$x_{4a} + x_{5a} + x_{4b} + x_{5b} \leq 1$$
$$x_{2a} + x_{4a} + x_{4b} \leq 1$$
$$x_{5a} + x_{3b} + x_{5b} \leq 1$$
$$x_{1c} + x_{1d} + x_{1e} \leq 1$$
$$x_{2d} + x_{1e} + x_{4e} \leq 1$$
$$x_{3c} + x_{3d} + x_{4e} + x_{6e} \leq 1$$
$$x_{4c} + x_{6e} \leq 1$$
$$x_{2a} + x_{4a} + x_{5a} = 1$$
$$x_{3b} + x_{4b} + x_{5b} = 1$$
$$x_{1c} + x_{3c} + x_{4c} = 1$$
$$x_{1d} + x_{2d} + x_{3d} = 1$$
$$x_{1e} + x_{4e} + x_{6e} = 1$$

Dari model matematika tersebut, dengan menggunakan metode simpleks didapat solusi terkini untuk RMP1 dengan nilai fungsi tujuan adalah 13 dan nilai variabel $x_{4a} = x_{3b} = x_{4c} = x_{3d} = x_{1e} = 1$ dan variabel yang lain bernilai nol.

Selanjutnya diberikan variabel dual w_{kh} , v_{hq}, u_c pada kendala-kendala RMP1 di atas. Sehingga didapatkan masalah dual untuk RMP1 berikut:

minimum $(2w_{11} + 2w_{12} + 2w_{13} + 2w_{14} + 2w_{21} + 2w_{22} + 2w_{23} + 2w_{24})$
$+ (v_{11} + v_{21} + v_{31} + v_{41} + v_{12} + v_{22} + v_{32} + v_{42}) +$
$(u_1 + u_2 + u_3 + u_4 + u_5)$

dk

$$w_{11}+w_{13}+v_{11}+v_{31}+u_1 \geq 1$$
$$w_{12}+w_{14}+v_{21}+v_{41}+u_1 \geq 4$$
$$w_{12}+w_{13}+v_{21}+v_{31}+u_1 \geq 3$$
$$w_{22}+w_{24}+v_{21}+v_{41}+u_2 \geq 1$$
$$w_{21}+w_{24}+v_{11}+v_{41}+u_2 \geq 3$$
$$w_{22}+w_{23}+v_{21}+v_{31}+u_2 \geq 1$$
$$w_{11}+v_{12}+u_3 \geq 1$$
$$w_{14}+v_{42}+u_3 \geq 2$$
$$w_{13}+v_{32}+u_3 \geq 0$$
$$w_{12}+v_{22}+u_4 \geq 0$$
$$w_{11}+v_{12}+u_4 \geq 2$$
$$w_{13}+v_{32}+u_4 \geq 1$$
$$w_{23}+w_{24}+v_{32}+v_{42}+u_5 \geq 1$$
$$w_{21}+w_{22}+v_{12}+v_{22}+u_5 \geq 4$$
$$w_{22}+w_{23}+v_{22}+v_{32}+u_5 \geq 3$$

Dari model dual di atas didapatkan solusi dual dengan nilai fungsi tujuan adalah 13 dan nilai variabel $w_{11}=w_{12}=w_{13}=w_{14}=w_{21}=w_{22}=w_{23}=w_{24}=0, (v_{11},v_{21,}v_{31,}v_{41})=(1,3,0,1), (v_{12},v_{22},v_{32},v_{42})=(1,3,0,1), (u_1,u_2,u_3,u_4,u_5)=(0,1,1,1,0)$.

Karena solusi primal RMP1 sama dengan solusi dual RMP1, maka solusi primal RMP1 merupakan solusi optimal untuk RMP1. Untuk mengetahui apakah solusi optimal RMP1 juga merupakan solusi optimal MP, akan dibentuk subproblem tiap MK:

- Subproblem MK c_1

$$w'_{ha} = \min_{k\epsilon K\,(c)} w_{kh} - s_{kha}$$
$$= (0\ 0\ 0\ 0) - (0\ 2\ 1\ 2) = (0\ \ -2\ \ -1\ \ -2)$$
$$v'_{ha} = v_{h1} = (1\ 3\ 0\ 1)$$
$$t_{ha} = w'_{ha} + v'_{ha} = (1\quad 1\ \ -1\ \ -1)$$

Alokasi waktu untuk MK c_1 adalah dua slot-waktu, sehingga minimumnya dipilih slot-waktu 3 dan 4 dengan nilai minimum $M_1 = (-1) + (-1) = -2$. Karena nilai $M_1 + u_1 = (-2) + (0) = -2 < 0$, maka solusi optimal belum terpenuhi untuk c_1, sehingga pola (0 0 1 1) akan menjadi kandidat untuk masuk ke RMP1.

- Subproblem MK c_2

$$w'_{hb} = \min_{k \epsilon K\,(c)} w_{kh} - s_{khb}$$
$$= (0\ \ 0\ \ 0\ \ 0) - (2\ \ 0\ \ 1\ \ 1) = (-2\ \ 0\ \ -1\ \ -1)$$
$$v'_{hb} = v_{h1} = (1\ 3\ 0\ 1)$$
$$t_{hb} = w'_{hb} + v'_{hb} = (-1\ \ \ 3\ \ -1\ \ 0)$$

Alokasi waktu untuk MK c_2 adalah dua slot-waktu, sehingga minimumnya dipilih slot-waktu 1 dan 3 dengan nilai minimum $M_2 = (-1) + (-1) = -2$. Karena nilai $M_2 + u_2 = (-2) + (1) = -1 < 0$, maka solusi optimal belum terpenuhi untuk c_2, sehingga pola (1 0 1 0) akan menjadi kandidat untuk masuk ke RMP1.

- Subproblem untuk c_3

$$w'_{hc} = \min_{k \epsilon K\,(c)} w_{kh} - s_{khc}$$
$$= \min\{w_{1h} - s_{1hc}; w_{2h} - s_{2hc}\}$$
$$= (-1\ \ -1\ \ \ 0\ \ -2)$$
$$v'_{hc} = v_{h2} = (1\ 3\ 0\ 1)$$
$$t_{hc} = w'_{hc} + v'_{hc} = (0\ \ 2\ \ 0\ \ -1)$$

Alokasi waktu untuk MK c_3 adalah satu slot-waktu, sehingga minimumnya dipilih slot-waktu 4 dengan nilai minimum $M_3 = -1$. Karena nilai $M_3 + u_3 = (-1) + (1) = 0$, maka solusi optimal sudah terpenuhi untuk c_3.

- Subproblem untuk c_4

$$w'_{hd} = \min_{k \epsilon K\,(c)} w_{kh} - s_{khd}$$
$$= \min\{w_{1h} - s_{1hd}; w_{2h} - s_{2hd}\}$$
$$= (-2\ \ \ 0\ \ -1\ \ -1)$$
$$v'_{hd} = v_{h2} = (1\ 3\ 0\ 1)$$
$$t_{hd} = w'_{hd} + v'_{hd} = (-1\ \ 3\ \ -1\ \ 0)$$

Alokasi waktu untuk MK c_4 adalah satu slot-waktu, sehingga minimumnya dipilih slot-waktu 1 atau 3 dengan nilai minimum $M_4 = -1$. Karena nilai $M_4 + u_4 = (-1) + (1) = 0$, maka solusi optimal sudah terpenuhi untuk c_4.

- Subproblem untuk c_5

$$w'_{he} = \min_{k \epsilon K(c)} w_{kh} - s_{khe}$$

$$= (-2 \ \ -2 \ -1 \ \ 0)$$

$$v'_{he} = v_{h2} = (1\ 3\ 0\ 1)$$

$$t_{he} = w'_{he} + v'_{he} = (-1 \ \ 1 \ -1 \ \ 1)$$

Alokasi waktu untuk MK c_5 adalah dua slot-waktu, sehingga minimumnya dipilih slot-waktu 1 dan 3 dengan nilai minimum $M_5 = -2$. Karena nilai $M_5 + u_5 = (-2) + (0) = -2$, maka solusi optimal belum terpenuhi untuk c_5, sehingga pola (1 0 1 0) menjadi kandidat untuk masuk ke RMP1.

Karena masih terdapat nilai *reduced cost* yang negatif, maka solusi optimal RMP1 belum merupakan solusi optimal untuk MP. *Reduced cost* untuk MK c_1 dan MK c_5 bernilai sama yaitu -2 (bernilai paling negatif), maka akan dipilih indeks terkecil yaitu MK c_1, sehingga pola (0 0 1 1) untuk MK c_1 dimasukkan ke RMP1.

Iterasi Kedua (RMP2)

Dengan menambahkan variabel x_{6a} ke RMP1, didapat RMP2 sebagai berikut:

maksimum $(x_{2a} + 3x_{4a} + 4x_{5a} + 3x_{6a}) + (3x_{3b} + x_{4b} + x_{5b}) + (x_{1c} + 0x_{3c} + 2x_{4c}) + (2x_{1d} + 0x_{2d} + x_{3d}) + (4x_{1e} + 3x_{4e} + x_{6e})$

dk

$$x_{2a} + x_{1c} + x_{1d} \leq 2$$

$$x_{4a} + x_{5a} + x_{2d} \leq 2$$

$$x_{2a} + x_{4a} + x_{6a} + x_{3c} + x_{3d} \leq 2$$

$$x_{5a} + x_{6a} + x_{4c} \leq 2$$

$$x_{3b} + x_{1e} \leq 2$$

$$x_{4b} + x_{5b} + x_{1e} + x_{4e} \leq 2$$

$$x_{4b} + x_{4e} + x_{6e} \leq 2$$

$$x_{3b} + x_{5b} + x_{6e} \leq 2$$
$$x_{2a} + x_{3b} \leq 1$$
$$x_{4a} + x_{5a} + x_{4b} + x_{5b} \leq 1$$
$$x_{2a} + x_{4a} + x_{6a} + x_{4b} \leq 1$$
$$x_{5a} + x_{6a} + x_{3b} + x_{5b} \leq 1$$
$$x_{1c} + x_{1d} + x_{1e} \leq 1$$
$$x_{2d} + x_{1e} + x_{4e} \leq 1$$
$$x_{3c} + x_{3d} + x_{4e} + x_{6e} \leq 1$$
$$x_{4c} + x_{6e} \leq 1$$
$$x_{2a} + x_{4a} + x_{5a} + x_{6a} = 1$$
$$x_{3b} + x_{4b} + x_{5b} = 1$$
$$x_{1c} + x_{3c} + x_{4c} = 1$$
$$x_{1d} + x_{2d} + x_{3d} = 1$$
$$x_{1e} + x_{4e} + x_{6e} = 1$$

Dari model matematika tersebut, dengan metode simpleks didapat solusi terkini untuk RMP2 dengan nilai fungsi tujuan adalah 13 dengan nilai variabel $x_{4a} = x_{3b} = x_{4c} = x_{1d} = x_{4e} = 1$ dengan variabel yang lain bernilai nol.

Selanjutnya diberikan variabel dual w_{kh} , v_{hq}, u_c pada kendala-kendala RMP2 di atas. Sehingga didapatkan masalah dual2 berikut:

minimum $(2w_{11} + 2w_{12} + 2w_{13} + 2w_{14} + 2w_{21} + 2w_{22} + 2w_{23} + 2w_{24}) + (v_{11} + v_{21} + v_{31} + v_{41} + v_{12} + v_{22} + v_{32} + v_{42}) + (u_1 + u_2 + u_3 + u_4 + u_5)$

dk

$$w_{11} + w_{13} + v_{11} + v_{41} + u_1 \geq 1$$
$$w_{12} + w_{14} + v_{21} + v_{41} + u_1 \geq 4$$
$$w_{12} + w_{13} + v_{21} + v_{31} + u_1 \geq 3$$
$$w_{13} + w_{14} + v_{31} + v_{41} + u_1 \geq 3$$
$$w_{22} + w_{24} + v_{21} + v_{24} + u_2 \geq 1$$
$$w_{21} + w_{24} + v_{11} + v_{41} + u_2 \geq 3$$
$$w_{22} + w_{23} + v_{21} + v_{31} + u_2 \geq 1$$
$$w_{11} + v_{12} + u_3 \geq 1$$

$$w_{14} + v_{42} + u_3 \geq 2$$
$$w_{13} + v_{32} + u_3 \geq 0$$
$$w_{12} + v_{22} + u_4 \geq 0$$
$$w_{11} + v_{12} + u_4 \geq 2$$
$$w_{13} + v_{32} + u_4 \geq 1$$
$$w_{23} + w_{24} + v_{32} + v_{42} + u_5 \geq 1$$
$$w_{21} + w_{22} + v_{12} + v_{22} + u_5 \geq 4$$
$$w_{22} + w_{23} + v_{22} + v_{32} + u_5 \geq 3$$

Dari model dual di atas didapatkan solusi dual untuk dual2 dengan nilai fungsi tujuan adalah 13 dengan nilai variabel $w_{11} = w_{12} = w_{13} = w_{14} = w_{21} = w_{22} = w_{23} = w_{24} = 0, (v_{11}, v_{21}, v_{31}, v_{41}) = (0,2,1,2), (v_{12}, v_{22}, v_{32}, v_{42}) = (1,3,0,1), (u_1, u_2, u_3, u_4, u_5) = (0,1,1,1,0)$.

Karena solusi primal RMP2 sama dengan solusi dualnya, maka solusi primal RMP2 merupakan solusi optimal untuk RMP2. Untuk mengetahui solusi terkini RMP2 akan menjadi solusi optimal MP, akan dibentuk subproblem tiap MK:

- Subproblem MK c_1

$$w'_{ha} = \min_{k \epsilon K(c)} w_{kh} - s_{kha}$$
$$= (0\ \ 0\ 0\ \ 0) - (0\ \ 2\ \ 1\ \ 2) = (0\ \ -2\ \ -1\ \ -2)$$
$$v'_{ha} = v_{h1} = (0\ 2\ 1\ 2)$$
$$t_{ha} = w'_{ha} + v'_{ha} = (0\quad 0\ \ 0\ \ \ 0)$$

Alokasi waktu untuk MK c_1 adalah dua slot-waktu, sehingga minimumnya dipilih di sembarang slot-waktu dengan nilai minimum $M_1 = (0) + (0) = 0$. Karena nilai $M_1 + u_1 = (0) + (0) = 0$, maka solusi optimal sudah terpenuhi untuk c_1.

- Subproblem MK c_2

$$w'_{hb} = \min_{k \epsilon K(c)} w_{kh} - s_{khb}$$
$$= (0\ \ 0\ \ 0\ \ 0) - (2\ \ 0\ \ 1\ \ 1) = (-2\ \ 0\ \ -1\ \ -1)$$
$$v'_{hb} = v_{h1} = (0\ 2\ 1\ 2)$$
$$t_{hb} = w'_{hb} + v'_{hb} = (-2\quad 2\ \ 0\ \ \ 1)$$

Alokasi waktu untuk MK c_2 adalah dua slot-waktu, sehingga minimumnya dipilih slot-waktu 1 dan 3 dengan nilai minimum $M_2 = (-2) + (0) = -2$. Karena nilai $M_2 + u_2 = (-2) + (1) = -1 < 0$, maka solusi optimal belum terpenuhi untuk c_2, sehingga pola (1 0 1 0) akan menjadi kandidat untuk masuk ke RMP2.

- Subproblem untuk c_3

$$\begin{aligned} w'_{hc} &= \min_{k\epsilon K(c)} w_{kh} - s_{khc} \\ &= \min\{w_{1h} - s_{1hc}; w_{2h} - s_{2hc}\} \\ &= (-1 \quad -1 \quad 0 \quad -2) \end{aligned}$$

$$v'_{hc} = v_{h2} = (1\ 3\ 0\ 1)$$

$$t_{hc} = w'_{hc} + v'_{hc} = (0 \quad 2 \quad 0 \quad -1)$$

Alokasi waktu untuk MK c_3 adalah satu slot-waktu, sehingga minimumnya dipilih slot-waktu 4 dengan nilai minimum $M_3 = -1$. Karena nilai $M_3 + u_3 = (-1) + (1) = 0$, maka solusi optimal sudah terpenuhi untuk c_3.

- Subproblem untuk c_4

$$\begin{aligned} w'_{hd} &= \min_{k\epsilon K(c)} w_{kh} - s_{khd} \\ &= \min\{w_{1h} - s_{1hd}; w_{2h} - s_{2hd}\} \\ &= (-2 \quad 0 \quad -1 \quad -1) \end{aligned}$$

$$v'_{hd} = v_{h2} = (1\ 3\ 0\ 1)$$

$$t_{hd} = w'_{hd} + v'_{hd} = (-1 \quad 3 \quad -1 \quad 0)$$

Alokasi waktu untuk MK c_4 adalah satu slot-waktu, sehingga minimumnya dipilih slot-waktu 1 atau 3 dengan nilai minimum $M_4 = -1$. Karena nilai $M_4 + u_4 = (-1) + (1) = 0$, maka solusi optimal sudah terpenuhi untuk c_4.

- Subproblem untuk c_5

$$\begin{aligned} w'_{he} &= \min_{k\epsilon K(c)} w_{kh} - s_{khe} \\ &= (-2 \quad -2 \quad -1 \quad 0) \end{aligned}$$

$$v'_{he} = v_{h2} = (1\ 3\ 0\ 1)$$

$$t_{he} = w'_{he} + v'_{he} = (-1 \quad 1 \quad -1 \quad 1)$$

Alokasi waktu untuk MK c_5 adalah dua slot-waktu, sehingga minimumnya dipilih slot-waktu 1 dan 3 dengan nilai minimum $M_5 = -2$. Karena nilai $M_5 +$

$u_5 = (-2) + (0) = -2$, maka solusi optimal belum terpenuhi untuk c_5, sehingga pola (1 0 1 0) menjadi kandidat untuk masuk ke RMP2.

Karena masih terdapat nilai *reduced cost* yang negatif, maka solusi optimal RMP2 belum merupakan solusi optimal untuk MP. *Reduced cost* untuk MK $c_5$bernilai -2 (bernilai paling negatif), maka pola (1 0 1 0) akan dimasukkan ke RMP2.

Iterasi Ketiga (RMP3)

Dengan menambahkan variabel x_{2e} ke RMP2, didapat RMP3 sebagai berikut:

$$\text{maksimum} \quad (x_{2a} + 3x_{4a} + 4x_{5a} + 3x_{6a}) + (3x_{3b} + x_{4b} + x_{5b}) + (x_{1c} + 0x_{3c} + 2x_{4c}) + (2x_{1d} + 0x_{2d} + x_{3d}) + (4x_{1e} + 3x_{2e} + 3x_{4e} + x_{6e})$$

dk

$$x_{2a} + x_{1c} + x_{1d} \le 2$$
$$x_{4a} + x_{5a} + x_{2d} \le 2$$
$$x_{2a} + x_{4a} + x_{6a} + x_{3c} + x_{3d} \le 2$$
$$x_{5a} + x_{6a} + x_{4c} \le 2$$
$$x_{3b} + x_{1e} + x_{2e} \le 2$$
$$x_{4b} + x_{5b} + x_{1e} + x_{4e} \le 2$$
$$x_{4b} + x_{2e} + x_{4e} + x_{6e} \le 2$$
$$x_{3b} + x_{5b} + x_{6e} \le 2$$
$$x_{2a} + x_{3b} \le 1$$
$$x_{4a} + x_{5a} + x_{4b} + x_{5b} \le 1$$
$$x_{2a} + x_{4a} + x_{6a} + x_{4b} \le 1$$
$$x_{5a} + x_{6a} + x_{3b} + x_{5b} \le 1$$
$$x_{1c} + x_{1d} + x_{1e} + x_{2e} \le 1$$
$$x_{2d} + x_{1e} + x_{4e} \le 1$$
$$x_{3c} + x_{3d} + x_{2e} + x_{4e} + x_{6e} \le 1$$
$$x_{4c} + x_{6e} \le 1$$
$$x_{2a} + x_{4a} + x_{5a} + x_{6a} = 1$$

$$x_{3b} + x_{4b} + x_{5b} = 1$$
$$x_{1c} \quad + x_{3c} + x_{4c} = 1$$
$$x_{1d} + x_{2d} + x_{3d} = 1$$
$$x_{1e} + x_{2e} + x_{4e} + x_{6e} = 1$$

Dari model matematika tersebut, dengan metode simpleks didapat solusi terkini untuk RMP3 dengan nilai fungsi tujuan adalah 13 dengan nilai variabel $x_{4a} = x_{3b} = x_{4c} = x_{1d} = x_{4e} = 1$ dan variabel yang lain bernilai nol.

Selanjutnya diberikan variabel dual w_{kh} , v_{hq}, u_c pada kendala-kendala RMP3 di atas. Sehingga didapatkan masalah dual3 berikut:

minimum $(2w_{11} + 2w_{12} + 2w_{13} + 2w_{14} + 2w_{21} + 2w_{22} + 2w_{23} + 2w_{24})$
$+ (v_{11} + v_{21} + v_{31} + v_{41} + v_{12} + v_{22} + v_{32} + v_{42}) +$
$(u_1 + u_2 + u_3 + u_4 + u_5)$

dk

$$w_{11} + w_{13} + v_{11} + v_{41} + u_1 \geq 1$$
$$w_{12} + w_{14} + v_{21} + v_{41} + u_1 \geq 4$$
$$w_{12} + w_{13} + v_{21} + v_{31} + u_1 \geq 3$$
$$w_{13} + w_{14} + v_{31} + v_{41} + u_1 \geq 3$$
$$w_{22} + w_{24} + v_{21} + v_{24} + u_2 \geq 1$$
$$w_{21} + w_{24} + v_{11} + v_{41} + u_2 \geq 3$$
$$w_{22} + w_{23} + v_{21} + v_{31} + u_2 \geq 1$$
$$w_{11} + v_{12} + u_3 \geq 1$$
$$w_{14} + v_{42} + u_3 \geq 2$$
$$w_{13} + v_{32} + u_3 \geq 0$$
$$w_{12} + v_{22} + u_4 \geq 0$$
$$w_{11} + v_{12} + u_4 \geq 2$$
$$w_{13} + v_{32} + u_4 \geq 1$$
$$w_{23} + w_{24} + v_{32} + v_{42} + u_5 \geq 1$$
$$w_{21} + w_{22} + v_{12} + v_{22} + u_5 \geq 4$$
$$w_{22} + w_{23} + v_{22} + v_{32} + u_5 \geq 3$$
$$w_{21} + w_{23} + v_{12} + v_{32} + u_5 \geq 3$$

Dari model dual3 di atas didapatkan solusi dual dengan nilai fungsi tujuan adalah 13 dengan nilai variabel $w_{11} = w_{12} = w_{13} = w_{14} = w_{21} = w_{22} = w_{23} = w_{24} = 0, (v_{11}, v_{21}, v_{31}, v_{41}) = (0,2,1,2), (v_{12}, v_{22}, v_{32}, v_{42}) = (2,2,1,0), (u_1, u_2, u_3, u_4, u_5) = (0,1,2,0,0)$.

Karena solusi primal RMP3 sama dengan solusi dualnya, maka solusi primal RMP3 merupakan solusi optimal untuk RMP3. Untuk mengetahui solusi terkini RMP3 akan menjadi solusi optimal MP, akan dibentuk subproblem tiap MK:

- Subproblem MK c_1

$$w'_{ha} = \min_{k \epsilon K(c)} w_{kh} - s_{kha}$$

$$= (0\ \ 0\ 0\ \ 0) - (0\ \ 2\ \ 1\ \ 2)\ = (0\ \ -2\ \ -1\ \ -2)$$

$$v'_{ha} = v_{h1} = (0\ 2\ 1\ 2)$$

$$t_{ha} = w'_{ha} + v'_{ha} = (0\quad 0\ \ 0\ \ 0)$$

Alokasi waktu untuk MK c_1 adalah dua slot-waktu, sehingga minimumnya dipilih di slot-waktu berapapun dengan nilai minimum $M_1 = (0) + (0) = 0$. Karena nilai $M_1 + u_1 = (0) + (0) = 0$, maka solusi optimal sudah terpenuhi untuk c_1.

- Subproblem MK c_2

$$w'_{hb} = \min_{k \epsilon K(c)} w_{kh} - s_{khb}$$

$$= (0\ \ 0\ \ 0\ \ 0) - (2\ \ 0\ \ 1\ \ 1)\ = (-2\ \ 0\ \ -1\ \ -1)$$

$$v'_{hb} = v_{h1} = (0\ 2\ 1\ 2)$$

$$t_{hb} = w'_{hb} + v'_{hb} = (-2\quad 2\ \ 0\ \ 1)$$

Alokasi waktu untuk MK c_2 adalah dua slot-waktu, sehingga minimumnya dipilih slot-waktu 1 dan 3 dengan nilai minimum $M_2 = (-2) + (0) = -2$. Karena $M_2 + u_2 = (-2) + (1) = -1 < 0$, maka solusi optimal belum terpenuhi untuk c_2, sehingga pola (1 0 1 0) akan menjadi kandidat untuk masuk ke RMP3.

- Subproblem untuk c_3

$$w'_{hc} = \min_{k \epsilon K(c)} w_{kh} - s_{khc}$$

$$= \min\{w_{1h} - s_{1hc}; w_{2h} - s_{2hc}\}$$

$$= (-1\ \ -1\ \ \ 0\ \ -2)$$

$$v'_{hc} = v_{h2} = (2\ 2\ 1\ 0)$$
$$t_{hc} = w'_{hc} + v'_{hc} = (1\ \ 1\ \ 1\ \ -2)$$

Alokasi waktu untuk MK c_3 adalah satu slot-waktu, sehingga minimumnya dipilih slot-waktu 4 dengan nilai minimum $M_3 = -2$. Karena nilai $M_3 + u_3 = (-2) + (2) = 0$, maka solusi optimal sudah terpenuhi untuk c_3.

- Subproblem untuk c_4

$$w'_{hd} = \min_{k \epsilon K\,(c)} w_{kh} - s_{khd}$$
$$= \min\{w_{1h} - s_{1hd}; w_{2h} - s_{2hd}\}$$
$$= (-2\ \ 0\ -1\ -1)$$
$$v'_{hd} = v_{h2} = (2\ 2\ 1\ 0)$$
$$t_{hd} = w'_{hd} + v'_{hd} = (0\ \ 2\ 0\ -1)$$

Alokasi waktu untuk MK c_4 adalah satu slot-waktu, sehingga minimumnya dipilih slot-waktu 4 dengan nilai minimum $M_4 = -1$. Karena nilai $M_4 + u_4 = (-1) + (0) = -1$, maka solusi optimal belum terpenuhi untuk c_4, sehingga pola (0 0 0 1) menjadi kandidat untuk masuk ke RMP3.

- Subproblem untuk c_5

$$w'_{he} = \min_{k \epsilon K\,(c)} w_{kh} - s_{khe}$$
$$= (-2\ \ -2\ -1\ \ 0)$$
$$v'_{he} = v_{h2} = (2\ 2\ 1\ 0)$$
$$t_{he} = w'_{he} + v'_{h1} = (0\ \ 0\ 0\ \ 0)$$

Alokasi waktu untuk MK c_5 adalah dua slot-waktu, sehingga minimumnya dipilih sembarang slot-waktu dengan nilai minimum $M_5 = 0$. Karena nilai $M_5 + u_5 = (0) + (0) = 0$, maka solusi optimal sudah terpenuhi untuk c_5.

Karena masih terdapat nilai *reduced cost* yang negatif maka solusi optimal RMP3 belum merupakan solusi optimal MP. *Reduced cost* untuk MK c_2 dan MK c_4 bernilai sama yaitu -1 (bernilai paling negatif), maka akan dipilih index terkecil yaitu pola (1 0 1 0) untuk MK c_2 akan dimasukan ke RMP3.

Iterasi Keempat (RMP4)

Dengan menambahkan variabel x_{2b} ke RMP3, didapat RMP4 sebagai berikut:

maksimum $(x_{2a} + 3x_{4a} + 4x_{5a} + 3x_{6a}) + (3x_{2b} + 3x_{3b} + x_{4b} + x_{5b})$
$+ (x_{1c} + 0x_{3c} + 2x_{4c}) + (2x_{1d} + 0x_{2d} + x_{3d}) +$
$(4x_{1e} + 3x_{2e} + 3x_{4e} + x_{6e})$

dk

$$x_{2a} + x_{1c} + x_{1d} \leq 2$$
$$x_{4a} + x_{5a} + x_{2d} \leq 2$$
$$x_{2a} + x_{4a} + x_{6a} + x_{3c} + x_{3d} \leq 2$$
$$x_{5a} + x_{6a} + x_{4c} \leq 2$$
$$x_{3b} + x_{1e} + x_{2e} \leq 2$$
$$x_{2b} + x_{4b} + x_{5b} + x_{1e} + x_{4e} \leq 2$$
$$x_{4b} + x_{2e} + x_{4e} + x_{6e} \leq 2$$
$$x_{2b} + x_{3b} + x_{5b} + x_{6e} \leq 2$$
$$x_{2a} + x_{2b} + x_{3b} \leq 1$$
$$x_{4a} + x_{5a} + x_{4b} + x_{5b} \leq 1$$
$$x_{2a} + x_{4a} + x_{6a} + x_{2b} + x_{4b} \leq 1$$
$$x_{5a} + x_{6a} + x_{3b} + x_{5b} \leq 1$$
$$x_{1c} + x_{1d} + x_{1e} + x_{2e} \leq 1$$
$$x_{2d} + x_{1e} + x_{4e} \leq 1$$
$$x_{3c} + x_{3d} + x_{2e} + x_{4e} + x_{6e} \leq 1$$
$$x_{4c} + x_{6e} \leq 1$$
$$x_{2a} + x_{4a} + x_{5a} + x_{6a} = 1$$
$$x_{3b} + x_{4b} + x_{5b} = 1$$
$$x_{1c} + x_{3c} + x_{4c} = 1$$
$$x_{1d} + x_{2d} + x_{3d} = 1$$
$$x_{1e} + x_{2e} + x_{4e} + x_{6e} = 1$$

Dari model matematika tersebut, dengan metode simpleks didapat solusi terkini dengan nilai fungsi tujuan untuk RMP4 adalah 14 dengan nilai variabel $x_{5a} = x_{2b} = x_{4c} = x_{3d} = x_{1e} = 1$ dan variabel yang lain bernilai nol.

Selanjutnya diberikan variabel dual w_{kh} , v_{hq}, u_c pada kendala-kendala RMP4 di atas. Sehingga didapatkan masalah dual4 berikut:

minimum $(2w_{11} + 2w_{12} + 2w_{13} + 2w_{14} + 2w_{21} + 2w_{22} + 2w_{23} + 2w_{24})$
$+ (v_{11} + v_{21} + v_{31} + v_{41} + v_{12} + v_{22} + v_{32} + v_{42}) +$
$(u_1 + u_2 + u_3 + u_4 + u_5)$

dk

$$w_{11} + w_{13} + v_{11} + v_{41} + u_1 \geq 1$$
$$w_{12} + w_{14} + v_{21} + v_{41} + u_1 \geq 4$$
$$w_{12} + w_{13} + v_{21} + v_{31} + u_1 \geq 3$$
$$w_{13} + w_{14} + v_{31} + v_{41} + u_1 \geq 3$$
$$w_{22} + w_{24} + v_{21} + v_{24} + u_2 \geq 1$$
$$w_{21} + w_{24} + v_{11} + v_{41} + u_2 \geq 3$$
$$w_{22} + w_{23} + v_{21} + v_{31} + u_2 \geq 1$$
$$w_{21} + w_{23} + v_{11} + v_{31} + u_2 \geq 3$$
$$w_{11} + v_{12} + u_3 \geq 1$$
$$w_{14} + v_{42} + u_3 \geq 2$$
$$w_{13} + v_{32} + u_3 \geq 0$$
$$w_{12} + v_{22} + u_4 \geq 0$$
$$w_{11} + v_{12} + u_4 \geq 2$$
$$w_{13} + v_{32} + u_4 \geq 1$$
$$w_{23} + w_{24} + v_{32} + v_{42} + u_5 \geq 1$$
$$w_{21} + w_{22} + v_{12} + v_{22} + u_5 \geq 4$$
$$w_{22} + w_{23} + v_{22} + v_{32} + u_5 \geq 3$$
$$w_{21} + w_{23} + v_{12} + v_{32} + u_5 \geq 3$$

Dari model dual4 di atas didapatkan solusi dual dengan nilai fungsi tujuan adalah 14 dengan nilai variabel $(w_{11}, w_{12}, w_{13}, w_{14}) = (0,0,0,1)$,

$$(w_{21}, w_{22}, w_{23}, w_{24}) = (0,0,0,0), (v_{11}, v_{21}, v_{31}, v_{41}) = (0,2,1,1),$$
$$(v_{12}, v_{22}, v_{32}, v_{42}) = (2,2,1,0), (u_1, u_2, u_3, u_4, u_5) = (0,2,1,0,0).$$

Karena nilai solusi primal RMP4 sama dengan solusi dualnya, maka solusi primal RMP4 merupakan solusi optimal untuk RMP4. Untuk mengetahui apakah solusi optimal RMP4 juga merupakan solusi optimal MP, akan dibentuk subproblem tiap MK:

- Subproblem MK c_1

$$w'_{ha} = \min_{k \epsilon K\,(c)} w_{kh} - s_{kha}$$

$$= (0\ \ 0\ 0\ \ 1)\text{-}(0\ \ 2\ \ 1\ \ 2) = (0\ \ \text{-}2\ \ \text{-}1\ \ \text{-}1)$$

$$v'_{ha} = v_{h1} = (0\ 2\ 1\ 1)$$

$$t_{ha} = w'_{ha} + v'_{ha} = (0 \quad 0\ \ 0\ \ \ 0)$$

Alokasi waktu untuk MK c_1 adalah dua slot-waktu, sehingga minimumnya dipilih sembarang slot-waktu dengan nilai minimum $M_1 = (0) + (0) = 0$. Karena nilai $M_1 + u_1 = (0) + (0) = 0$, maka solusi optimal sudah terpenuhi untuk c_1.

- Subproblem MK c_2

$$w'_{hb} = \min_{k \epsilon K\,(c)} w_{kh} - s_{khb}$$

$$= (0\ \ 0\ \ 0\ \ 0) - (2\ \ 0\ \ 1\ \ 1) = (\text{-}2\ \ 0\ \ \text{-}1\ \ \text{-}1)$$

$$v'_{hb} = v_{h1} = (0\ 2\ 1\ 1)$$

$$t_{hb} = w'_{hb} + v'_{hb} = (-2 \quad 2\ \ 0\ \ \ 0)$$

Alokasi waktu untuk MK c_2 adalah dua slot-waktu, sehingga minimumnya dipilih slot-waktu 1 dan 3 dengan nilai minimum $M_2 = (-2) + (0) = -2$. Karena nilai $M_2 + u_2 = (-2) + (2) = 0$, maka solusi optimal sudah terpenuhi untuk c_2.

- Subproblem untuk c_3

$$w'_{hc} = \min_{k \epsilon K\,(c)} w_{kh} - s_{khc}$$

$$= \min\{w_{1h} - s_{1hc}; w_{2h} - s_{2hc}\}$$

$$= (-1 \quad -1 \quad 0 \quad -1)$$

$$v'_{hc} = v_{h2} = (2\ 2\ 1\ 0)$$

$$t_{hc} = w'_{hc} + v'_{hc} = (1\ \ 1\ \ 1 \quad -1)$$

Alokasi waktu untuk MK c_3 adalah satu slot-waktu, sehingga minimumnya dipilih slot-waktu 4 dengan nilai minimum $M_3 = -1$. Karena nilai $M_3 + u_3 = (-1) + (1) = 0$, maka solusi optimal sudah terpenuhi untuk c_3.

- Subproblem untuk c_4

$$w'_{hd} = \min_{k \epsilon K\,(c)} w_{kh} - s_{khd}$$

$$= \min\{w_{1h} - s_{1hd}; w_{2h} - s_{2hd}\}$$

$$= (-2 \quad 0 \ -1\ 0)$$

$$v'_{hd} = v_{h2} = (2\ 2\ 1\ 0)$$

$$t_{hd} = w'_{hd} + v'_{hd} = (0\ \ 2\ 0\ 0)$$

Alokasi waktu untuk MK c_4 adalah satu slot-waktu, sehingga minimumnya dipilih slot-waktu 1 atau 3 atau 4 dengan nilai minimum $M_4 = 0$. Karena nilai $M_4 + u_4 = (0) + (0) = 0$, maka solusi optimal sudah terpenuhi untuk c_4.

- Subproblem untuk c_5

$$w'_{he} = \min_{k \epsilon K\,(c)} w_{kh} - s_{khe}$$

$$= (-2\ \ -2\ \ -1\ \ 0)$$

$$v'_{he} = v_{h2} = (2\ 2\ 1\ 0)$$

$$t_{he} = w'_{he} + v'_{he} = (0\ \ 0\ 0\ \ 0)$$

Alokasi waktu untuk MK c_5 adalah dua slot-waktu, sehingga minimumnya dipilih sembarang slot-waktu dengan nilai minimum $M_5 = 0$. Karena nilai $M_5 + u_5 = (0) + (0) = 0$, maka solusi optimal sudah terpenuhi untuk c_5.
Karena *reduced cost* tiap variabel tidak ada yang bernilai negatif dan solusi optimal RMP4 memenuhi kondisi integral,yaitu $x_{5a} = x_{2b} = x_{4c} = x_{3d} = x_{1e} = 1$ dan variabel yang lain bernilai nol, maka solusi optimal RMP4 juga merupakan solusi *master problem*.

Dengan kata lain, untuk MK c_1 pola yang dipilih adalah pola ke-5, yaitu slot-waktu 2 dan 4, untuk MK c_2 adalah pola ke-2, yaitu slot-waktu 1 dan 3, untuk MK c_3 adalah pola ke-4, yaitu slot-waktu 4, untuk MK c_4 adalah pola ke-3, yaitu slot-waktu 3, untuk MK c_5 adalah pola ke-1, yaitu slot-waktu 1 dan 2.
Sehingga penjadwalan 5 MK untuk contoh di atas adalah seperti tercantum di Tabel 3.2 berikut:

Tabel 3.2: Hasil penjadwalan 5 MK

Mata Kuliah	**Slot-waktu**			
	1	2	3	4
1		$c_1\ (R = k_1)$		$c_1\ (R = k_1)$
2	$c_2\ (R = k_2)$		$c_2\ (R = k_2)$	
3				$c_3\ (R = k_1)$
4			$c_4\ (R = k_1)$	
5	$c_5\ (R = k_2)$	$c_5\ (R = k_2)$		

ket: $n_{k_1} = n_{k_2} = 2$

Dari Tabel 3.2 di atas dapat dilihat bahwa setiap MK ditempatkan di ruangan yang cocok untuk MK tersebut. Juga tidak ada bentrokan waktu antara mata kuliah-mata kuliah dalam satu grup. Selain itu, nilai pemilihan slot-waktu untuk MK 1, yaitu slot-waktu ke-2 dan ke-4 bernilai 2. Untuk MK 2, yaitu slot-waktu ke-1 bernilai 2 dan slot-waktu ke-3 bernilai 1. Untuk MK 3, yaitu slot-waktu ke-4 bernilai 2. Untuk MK 4, yaitu slot-waktu ke-3 bernilai 1. Untuk MK 5, yaitu slot-waktu ke-1 dan ke-2 bernilai 2. Karena memenuhi hampir seluruh pemilihan slot-waktu yang diinginkan, maka dapat disimpulkan jadwal yang didapatkan untuk penjadwalan 5 MK di atas adalah optimal.

BAB 4

APLIKASI PADA PENJADWALAN MATA KULIAH DI DEPARTEMEN MATEMATIKA UNIVERSITAS INDONESIA

Pada bab ini akan dibahas tentang pendekatan *column generation* untuk penjadwalan mata kuliah (MK) yang telah dibahas sebelumnya yang diaplikasikan di Departemen Matematika Universitas Indonesia dengan menggunakan bantuan software matlab R2009a.

Penjadwalan kuliah di Departemen Matematika UI melibatkan beberapa komponen. Diantaranya ruang kuliah, slot-waktu, dan kelompok mata kuliah. Dosen dapat mengajar lebih dari satu MK dan satu MK dapat diajar oleh lebih dari satu dosen. Tetapi seorang dosen tidak dibolehkan mengajar lebih dari satu MK dalam waktu bersamaan. Oleh karena itu, mata kuliah-mata kuliah tersebut digabungkan dalam satu kelompok. Tersedia 8 ruangan dengan masing-masing kapasitas yang berbeda-beda. Disini penulis membagi menjadi 2 jenis ruangan, yaitu besar dan kecil.

Pada tugas akhir ini, akan dijadwalkan MK pada Semester Genap dengan MK yang dijadwalkan terdiri dari 24 MK, 20 slot-waktu, 2 jenis ruangan dengan masing-masing untuk ruangan kecil (k_1) berjumlah 5 atau $n_{k_1} = 5$ dan ruangan besar (k_2) berjumlah 3 atau $n_{k_2} = 3$, dan 3 kelompok MK, yaitu kelompok MK untuk mahasiswa semester 2, semester 4, dan semester 6. Untuk keterangan setiap MK yang ditempatkan di grup dan ruangan tertentu juga slot-waktu yang dibutuhkan setiap MK disajikan di Tabel 4.1 berikut.

Tabel 4.1: Data mata kuliah Semester Genap

No	Mata Kuliah	Slot-waktu yang dibutuhkan	Jenis Ruangan	Grup
1	Matematika Dasar II	2	k_2	1
2	Aljabar Linier Elementer	2	k_2	1
3	Statistika Elementer	2	k_2	1
4	Matematika Diskrit I	1	k_1	1
5	Metode Numerik	2	k_1	1

6	Matematika Dasar IV	2	k_2	2
7	Aljabar I	2	k_2	2
8	Statistika Matematika II	2	k_2	2
9	Analisi I	2	k_2	2
10	Statistika Pengendali Mutu	2	k_1	2
11	Pemrograman Linear	2	k_1	2
12	Matematika Numerik I	2	k_1	2
13	Matematika Keuangan	2	k_1	2
14	Geometri	2	k_1	2
15	Pemodelan Matematika	2	k_1	2
16	Model Linear II	2	k_1	3
17	Teori Ekonomi Keuangan	1	k_1	3
18	Pemrograman Dinamik	2	k_1	3
19	PDP &Syarat Batas	2	k_1	3
20	Komputasi Saintifik	2	k_1	3
21	Teori Komputasi	2	k_1	3
22	Runtun Waktu	2	k_2	3
23	Teori Ukur dan Integrasi	2	k_1	3
24	Metode Penelitian	1	k_1	3

Dalam skripsi ini diasumsikan tidak terdapat MK yang diajarkan oleh dosen yang sama. Nilai pemilihan apabila dosen berkeinginan untuk mengajar di slot-waktu tertentu adalah 2, jika antara bisa dan tidak, maka nilai 1, dan nilainya 0 jika tidak berkeinginan mengajar di slot-waktu tertentu. Misalkan pemilihan untuk setiap MK yang diajarkannya sebagai berikut:

Tabel 4.2: Nilai pemilihan slot-waktu oleh dosen

	Mata Kuliah 1 2 3 ……………………………………24
Senin	
08.00-10.00	0 2 1 1 0 0 1 1 1 2 1 1 1 0 1 1 1 1 2 1 1 1 0 1
10.00-12.00	1 0 1 2 1 1 0 2 0 1 1 1 1 1 0 1 0 1 1 0 0 1 2 0
13.00-15.00	1 1 0 1 1 2 1 0 1 1 0 1 1 1 1 0 1 1 1 1 1 2 1 1
15.00-17.00	0 0 0 0 1 0 0 0 0 0 0 2 0 0 1 2 0 0 0 0 0 0 0 0
Selasa	
08.00-10.00	0 1 1 1 0 0 2 1 1 1 1 1 1 0 1 1 1 1 2 1 1 1 0 1
10.00-12.00	2 0 1 0 0 1 0 1 0 2 0 0 1 0 0 1 0 2 1 0 0 1 1 0
13.00-15.00	1 1 2 1 1 1 1 0 1 1 2 1 1 1 1 0 1 1 1 1 2 1 1 1
15.00-17.00	0 0 0 0 2 0 0 0 2 0 0 0 0 0 1 0 0 0 0 2 0 0 0 0
Rabu	
08.00-10.00	0 1 1 1 1 0 1 1 1 1 1 1 1 0 0 1 1 1 0 1 1 0 0 1
10.00-12.00	1 0 1 0 1 1 0 2 0 1 1 0 1 0 0 1 2 1 1 0 0 0 1 0
13.00-15.00	1 1 0 1 1 1 1 0 1 1 0 1 1 1 2 0 1 1 1 1 2 1 1 1
15.00-17.00	0 1 0 0 2 0 0 0 0 0 0 0 0 2 1 0 0 0 0 1 0 1 0 0
Kamis	
08.00-10.00	0 2 1 0 0 1 2 1 1 1 1 1 1 0 1 2 1 1 0 1 0 1 0 1
10.00-12.00	1 0 1 0 1 2 0 1 0 1 1 0 1 2 0 1 0 2 1 0 0 1 1 0
13.00-15.00	1 1 0 1 1 1 1 0 1 1 0 2 1 1 1 0 1 1 1 1 1 1 2 1
15.00-17.00	0 0 0 0 2 0 0 0 0 0 0 0 2 0 1 0 2 1 0 0 0 0 0 0
Jum'at	
08.00-10.00	0 1 0 1 0 0 1 1 2 1 1 1 1 0 1 1 1 1 0 0 1 0 0 2
10.00-12.00	2 0 0 0 0 1 0 1 0 1 1 0 2 1 0 1 0 1 1 2 0 0 1 0
13.00-15.00	1 1 2 1 1 1 1 0 1 1 2 0 1 1 1 0 1 1 1 0 1 2 1 1
15.00-17.00	0 0 0 0 1 0 0 0 0 0 0 0 0 0 2 0 0 0 0 0 0 0 0 0

Dengan menggunakan program yang terdapat di lampiran, didapatkan solusi optimal dicapai saat iterasi ke-27 dengan waktu komputasi yang kurang dari 15 menit dan nilai fungsi tujuan adalah 88, sehingga didapatkan pola penjadwalan untuk setiap MK sebagai berikut.

$x_{94,1}$	: 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0
$x_{179,2}$	: 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
$x_{80,3}$	: 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0
$x_{2,4}$	: 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
$x_{33,5}$	: 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0
$x_{144,6}$	: 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
$x_{117,7}$	: 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0
$x_{164,8}$	: 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0
$x_{70,9}$	: 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0
$x_{186,10}$	: 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
$x_{80,11}$	: 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0
$x_{126,12}$	: 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
$x_{9,13}$	: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0
$x_{35,14}$	: 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0
$x_{37,15}$	: 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1
$x_{128,16}$	: 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
$x_{16,17}$	: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0
$x_{98,18}$	: 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0
$x_{187,19}$	: 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
$x_{69,20}$	: 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0
$x_{88,21}$	: 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0
$x_{138,22}$	: 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0
$x_{159,23}$	: 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
$x_{17,24}$	: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0

variabel keputusan $x_{94,1}$ menyatakan bahwa untuk MK ke-1 pola optimalnya adalah pola ke-94 yaitu slot-waktu ke-6 dan ke-18, berarti hari Selasa jam 10.00-12.00 dan hari Jum’at jam 10.00-12.00, variabel keputusan $x_{17,24}$

menyatakan bahwa untuk MK ke-24 pola optimalnya adalah pola ke-17 yaitu slot-waktu ke-17, berarti hari Jum'at jam 08.00-10.00 , seterusnya untuk variabel keputusan lainnya. Sehingga dari hasil di atas diperoleh penjadwalan MK Semester Genap di Departemen Matematika seperti yang tercantum pada Tabel di bawah.

	Ruangan							
	D109	D201	D303	D304	D402	D108	D401	D403
Senin								
8.00-10.00	PDP&Syarat Batas		Statistik Pengendali Mutu			Aljabar Linier Elementer		
10.00-12.00	Matematika Diskrit I				Teori Ukur dan Integrasi		Statistik Matematika II	
13.00-15.00							Matematika Dasar IV	Runtun Waktu
15.00-17.00			Matematika Numerik I	Model Linier I				
Selasa								
8.00-10.00	PDP&Syarat Batas						Aljabar I	
10.00-12.00			Statistik Pengendali Mutu	Pemrograma Dinamik		Matematika Dasar II		
13.00-15.00			Pemrograman Linier	Teori Komputasi		Statistik Elementer		
15.00-17.00	Komputasi Saintifik							Analis I
Rabu								
8.00-10.00							Aljabar I	
10.00-12.00							Statistik Matematika II	
13.00-15.00			Pemodelan Matematika	Teori Komputasi				
15.00-17.00	Metode Numerik		Geometri					
Kamis								
8.00-10.00				Model Linier II		Aljabar Linier Elementer	Matematika Dasar IV	
10.00-12.00			Geometri	Pemrograman Dinamik				
13.00-15.00			Matematika Numerik I		Teori Ukur dan Integrasi			
15.00-17.00	Metode Numerik		Matematika Keuangan	Teori Ekonomi Keuangan				
Jumat								
8.00-10.00					Metode Penelitian			Analis I
10.00-12.00	Komputasi Saintifik		Matematika Keuangan			Matematika Dasar II		
13.00-15.00			Pemrograman Linier			Statistik Elementer		Runtun Waktu
15.00-17.00			Pemodelan Matematika					

BAB 5
KESIMPULAN

Dari pembahasan bab-bab sebelumnya dapat disimpulkan bahwa penjadwalan mata kuliah dapat dimodelkan menjadi masalah program linier dan pendekatan *column generation* cukup efisien dalam menyelesaikan masalah penjadwalan mata kuliah tersebut karena tidak perlu dicari semua pola penjadwalan yang mungkin. Karena banyaknya pola penjadwalan yang sangat besar, maka cukup mengambil subhimpunan dari banyaknya pola tersebut. Kemudian pola penjadwalan baru ditambahkan hanya saat diperlukan.

Dari hasil aplikasi pendekatan *column generation* di masalah penjadwalan mata kuliah di Departemen Matematika, diperoleh jadwal yang optimal pada iterasi ke-27 dengan nilai fungsi tujuan adalah 88. Walaupun dalam proses pencarian jadwal dengan waktu komputasi yang cukup lama, yaitu kurang lebih 15 menit. Di samping itu, dari hasil yang didapat bahwa pemilihan waktu oleh dosen terpenuhi semaksimal mungkin dimana setiap mata kuliah yang didapat memiliki nilai pemilihan slot-waktu bernilai 2. Hal ini berarti, keinginan semua dosen untuk mengajar pada waktu yang diinginkan dapat terpenuhi.

DAFTAR PUSTAKA

Barnhart C., E.L. Johnson, G.L. Nemhauser, M.W.P. Savelsbergh and P. Vance. (1998). *Branch-and-price: Column generation for solving huge integer problems*, Operations Research, p. 316-329.

Burhan, H. (2005). Pendekatan *Column Generation* pada Masalah Penjadwalan Awak Bis. Thesis: Bandung, Matematika, FMIPA, Institut Teknologi Bandung.

Desrosiers, J dan Lubbecke, M.E. (2003). *A Primer in Column Generation. Technische Universitat Berlin*. p. 48

Hillier, F.S, Lieberman, G.J, *Introduction to Operation Research*, 6th edition, McGraw-Hill.

Lubbecke, M.E dan J. Desrosiers. (2002). *Selected Topics in Column Generation*. Les Cahiers du GERAD G-2002, HEC Montreal.

Mahayekti, H. (2007). Pendekatan Metode *Column Generation* pada *Cutting Stock Problem*. Skripsi: Depok, Departemen Matematika, FMIPA, Universitas Indonesia.

Papoutsis K., C. Valouxis and E. Housos,. (2003). *A column generation approach for the timetabling problem of Greek high schools*. J. of the Operational Research Society, p. 230-238.

Qualizza, A., dan P. Serafini. *A Column Generation Scheme for Faculty Timetabling*. University of Udine, Italy.

Wu, N., dan Coppins, R. (1981). *Linear Programming and Extenstions*. McGraw-Hill Series in Industrial Engineering and Management Science.

Lampiran:

IMPLEMENTASI PROGRAM DENGAN SOFTWARE MATLAB R2009a

```
function colgen_v1_2
clear; clc;
% INISIALISASI
% ---------------------------------------------------------------
% Menggunakan algoritma simplex untuk solve masalah LP
options = optimset('LargeScale', 'off', 'Simplex', 'on',
'Display', 'off');
% File Input
%f_input = input('Masukkan nama file input: ', 's');
fid = fopen('semest_genap.txt', 'r');
% Banyaknya Slot Waktu
H = fscanf(fid, '%d', [1,1]);
% Banyaknya MK
MK = fscanf(fid, '%d', [1,1]);
% Slot waktu untuk tiap MK
d = fscanf(fid, '%d', [1,MK]);
% Grup MK
jg = fscanf(fid, '%d', [1,1]);
C = zeros(jg,MK);
for i = 1 : jg
    C(i,:) = fscanf(fid, '%d', [1,MK]);
end
% Banyak kelas dan jenis kelas untuk tiap MK
jk = fscanf(fid, '%d', [1,1]);
nk = fscanf(fid, '%d', [1,jk]);
K = zeros(jg,MK);
for i = 1 : jk
    K(i,:) = fscanf(fid, '%d', [1,MK]);
end
%  Pemilihan tiap MK
S = zeros(H,MK);
for i = 1 : H
    S(i,:) = fscanf(fid, '%d', [1,MK]);
end
% Pola MK 2 slot-waktu
comb = combnk(1:H,2);
Pola_2t = zeros(H,size(comb,1));
i = size(comb,1);
j = 1;
while i >= 1
    Pola_2t(comb(i,1),j) = 1;
    Pola_2t(comb(i,2),j) = 1;
    i = i - 1;
    j = j + 1;
end
% Pola MK 1 slot-waktu
comb = combnk(1:H,1);
Pola_1t = zeros(H,size(comb,1));
for i = 1 : size(comb,1)
    Pola_1t(comb(i,1),i) = 1;
end
% ------------------------------------------
% Pembentukan Fungsi Objektif Master Problem
```

```
% ------------------------------------------
jvmp = 0;
for i = 1 : MK
    if d(i) == 1
        temp2 = size(Pola_1t,2);
    else
        temp2 = size(Pola_2t,2);
    end
    jvmp = temp2 + jvmp;
end
f = zeros(1,jvmp);
count = 1;
for i = 1 : MK
    if d(i) == 1
        PMK = size(Pola_1t,2);
        pola = Pola_1t;
    else
        PMK = size(Pola_2t,2);
        pola = Pola_2t;
    end
    for j = 1 : PMK
        temp = find(pola(:,j) == 1);
        if size(temp,1) == 1
            f(count) = S(temp,i);
        else
            f(count) = S(temp(1),i) + S(temp(2),i);
        end
        count = count + 1;
    end
end
% -------------------------------------------------
% Pembentukan Kendala Pertidaksamaan Master Problem
% -------------------------------------------------
A = zeros(((jk*H)+(jg*H)),size(f,2));
b = zeros(size(A,1),1);
% Kendala berdasarkan banyak mata kuliah dan slot-waktu
count = 1;
for i = 1 : jk
    for j = 1 : H
        start = 0;
        for k = 1 : MK
            if d(k) == 1
                PMK = size(Pola_1t,2);
                pola = Pola_1t;
            else
                PMK = size(Pola_2t,2);
                pola = Pola_2t;
            end
            if K(i,k) == 1
                A(count,(start+1):(PMK+start)) = pola(j,:);
            end
            start = PMK + start;
        end
        b(count) = nk(i);
        count = count + 1;
    end
end
% Kendala berdasarkan grup mata kuliah dan slot-waktu
for i = 1 : jg
```

```
    for j = 1 : H
        start = 0;
        for k = 1 : MK
            if d(k) == 1
                PMK = size(Pola_1t,2);
                pola = Pola_1t;
            else
                PMK = size(Pola_2t,2);
                pola = Pola_2t;
            end
            if C(i,k) == 1
                A(count,(start+1):(PMK+start)) = pola(j,:);
            end
            start = PMK + start;
        end
        b(count) = 1;
        count = count + 1;
    end
end
% --------------------------------------------
% Pembentukan Kendala Persamaan Master Problem
% --------------------------------------------
Aeq = zeros(MK,size(f,2));
beq = zeros(size(Aeq,1),1);
start = 0;
for i = 1 : MK
    if d(i) == 1
        PMK = size(Pola_1t,2);
    else
        PMK = size(Pola_2t,2);
    end
    Aeq(i,(start+1):(PMK+start)) = ones(1,PMK);
    beq(i) = 1;
    start = PMK + start;
end
% --------------------------------------------
% Batas Bawah dan Atas Variabel Master Problem
% --------------------------------------------
lb = zeros(1,size(f,2));
ub = ones(1,size(f,2));

% -------------------------------
% Pembentukan Fungsi Objektif RMP
% -------------------------------
p = 3;  % Banyaknya pola yang dipilih untuk tiap MK
fRMP = zeros(size(f));
temp3 = fRMP;
start = 0;
count = 0;
for i = 1 : MK
    if d(i) == 1
        PMK = size(Pola_1t,2);
    else
        PMK = size(Pola_2t,2);
    end
    [temp pos] = sort(f((start+1):(PMK+start)), 'descend');
    for j = 1 : p
        fRMP(start+pos(j)) = f(start+pos(j));
    end
```

```
        temp2 = find(temp(1:p) == 0);
        if ~isempty(temp2)
            for j = 1 : length(temp2)
                count = count + 1;
                temp3(count) = start + pos(temp2(j));
            end
        end
        start = PMK + start;
    end
    addv = temp3(1:count);
    % -----------------------
    % Pembentukan Kendala RMP
    % -----------------------
    ARMP = zeros(size(A));
    bRMP = b;
    AeqRMP = zeros(size(Aeq));
    beqRMP = beq;

    disp('Fungsi Objektif Master Problem:');
    disp(f);
    disp('==========================================');
    disp(' ');

    % ---------------------------------------------------------------
    % ITERASI UTAMA PEMBANGKITAN KOLOM
    % ---------------------------------------------------------------
    basis = zeros(MK,3);
    nul = find(fRMP == 0);
    basis(1,:) = [-1 nul(1) 0];
    counter = 0;
    [p q] = sort(basis(:,1));
    t = 1;
    while p(t) < 0
        while p(t) < 0 && fRMP(basis(q(t),2)) ~= 0 && t ~=
    size(basis,1)
            t = t + 1;
        end
        if (t == size(basis,1) && fRMP(basis(q(t),2)) ~= 0) || p(t) >=
    0
            disp(' ');
            disp('Program selesai: semua basis yang negatif sudah
    terdapat semua di fungsi objektif RMP');
            error('Program terminated');
        end
        % 1. Pembentukan RMP
        % Fungsi objektif RMP
        fRMP(basis(q(t),2)) = basis(q(t),3);
        % Kendala pertidaksamaan RMP (ARMP)
        temp = find(fRMP ~= 0);
        for j = 1 : size(A,1)
            for i = 1 : length(temp)
                if A(j,temp(i)) == 1
                    ARMP(j,temp(i)) = 1;
                end
            end
            for z = 1 : length(addv)
                if A(j,addv(z)) == 1
                    ARMP(j,addv(z)) = 1;
```

```
            end
        end
    end
    % Kendala persamaan RMP (AeqRMP)
    for j = 1 : size(Aeq,1)
        for i = 1 : length(temp)
            if Aeq(j,temp(i)) == 1
                AeqRMP(j,temp(i)) = 1;
            end
        end
        for z = 1 : length(addv)
            if A(j,addv(z)) == 1
                AeqRMP(j,addv(z)) = 1;
            end
        end
    end
    % -------------------
    % 2. Solve RMP
    [x, fval] = linprog(-
fRMP,ARMP,bRMP,AeqRMP,beqRMP,lb,ub,[],options);
    % -------------------

    % 3. Pembentukan dual RMP
    fd = [bRMP;beqRMP]';
    Ad = [ARMP;AeqRMP]';
    bd = fRMP';
    % -------------------
    % 4. Solve dual RMP
    [xd, fdval] = dsimplex('min',fd,Ad,bd);
    % -------------------

    % Subproblem masing-masing MK
    basis = zeros(MK,3);
    for i = 1 : MK
        % 5. Pembentukan subproblem
        % Penentuan nilai w
        pos = find(K(:,i) == 1);
        w = xd(((pos-1)*H+1):(pos*H));
        % Penentuan nilai v
        pos = find(C(:,i) == 1);
        strt = jk*H;
        v = xd((strt+(((pos-1)*H)+1)):(strt+(H*pos)));
        % Hitung nilai w', v', dan t
        w_ = w - S(:,i);
        v_ = v;
        t = w_ + v_;
        % 6. Solve subproblem
        curr_patt = zeros(H,1);
        [temp pos] = sort(t);
        M = 0;
        for j = 1 : d(i)
            M = temp(j) + M;
            curr_patt(pos(j)) = 1;
        end
        % Update basis
        selector = M + xd(jk*H+jg*H+i);
        basis(i,1) = selector;
        if selector < 0
```

```
            if d(i) == 1
                for j = 1 : size(Pola_1t,2)
                    if Pola_1t(:,j) == curr_patt
                        break
                    end
                end
            else
                for j = 1 : size(Pola_2t,2)
                    if Pola_2t(:,j) == curr_patt
                        break
                    end
                end
            end
            temp = 0;
            for r = 1 : (i-1)
                if d(r) == 1
                    temp2 = size(Pola_1t,2);
                else
                    temp2 = size(Pola_2t,2);
                end
                temp = temp2 + temp;
            end
            temp = temp + j;
            basis(i,2) = temp;
            basis(i,3) = f(temp);
        end
    end
    % -------------------
    [p q] = sort(basis(:,1));
    t = 1;
    % Penghitung iterasi
    counter = counter + 1;

    disp('Iterasi Ke:');
    disp(counter);
    disp('Fungsi Objektif RMP:');
    disp(fRMP);
    disp('Variabel RMP:');
    disp(x');
    disp('Nilai Fungsi Objektif RMP:');
    disp(-fval);
    disp('Variabel Dual RMP:');
    disp(xd');
    disp('Nilai Fungsi Objektif Dual RMP:');
    disp(fdval);
    disp('Basis:');
    disp(basis);
    disp('===========================================');
    disp(' ');
end
% ---------------------------------------------------------------
pos = find(x == 1);
count = 1;
counter = 1;
for i = 1 : MK
    if d(i) == 1
        PMK = size(Pola_1t,2);
        pola = Pola_1t;
    else
```

```
        PMK = size(Pola_2t,2);
        pola = Pola_2t;
    end
    for j = 1 : PMK
        if count <= size(pos,1)
        if counter == pos(count)
            fprintf('x%d%d : ',j,i);
            disp(pola(:,j)');
            count = count + 1;
        end
        end
        counter = counter + 1;
    end
end
end




function [x,z] = dsimplex(type, c, A, b)
% The Dual Simplex Algorithm for solving the LP problem
 xz
%                     min (max) z = c*x
%                   Subject to  Ax >= b
%                                x >= 0
%
if type == 'min'
   mm = 0;
else
   mm = 1;
   c = -c;
end
A = -A;
b = -b(:);
c = c(:)';
[m, n] = size(A);
A = [A eye(m) b];
A = [A;[c zeros(1,m+1)]];
subs = n+1:n+m;
[bmin, row] = Br(b);
tbn = 0;
while ~isempty(bmin) && bmin < 0 && abs(bmin) > eps
   if A(row,1:m+n) >= 0
      varargout(1)={subs(:)};
      varargout(2)={A};
      varargout(3) = {zeros(n,1)};
      varargout(4) = {0};
      return
   end
   col = MRTD(A(m+1,1:m+n),A(row,1:m+n));
   subs(row) = col;
   A(row,:)= A(row,:)/A(row,col);
      for i = 1:m+1
      if i ~= row
         A(i,:)= A(i,:)-A(i,col)*A(row,:);
      end
   end
   tbn = tbn + 1;
```

```
   [bmin, row] = Br(A(1:m,m+n+1));
end
x = zeros(m+n,1);
x(subs) = A(1:m,m+n+1);
x = x(1:n);
if mm == 0
   z = -A(m+1,m+n+1);
else
   z = A(m+1,m+n+1);
end
```

```
function col = MRTD(a, b)

% The Maximum Ratio Test performed on vectors a and b.
% This function is called from within the function dsimplex.
% Output parameter:
% col - index of the pivot column.

m = length(a);
c = 1:m;
a = a(:);
b = b(:);
l = c(b < 0);
[mi, col] = max(a(l)./b(l));
col = l(col);
```