

PENERAPAN AUTO GENERATED PROBABILISTIC COMBINATION (AGPC) PADA PROSES PEMBENTUKAN JADWAL RENCANA STUDY (JRS)

Untung Rahardja¹, Hidayati²

Jurusan Sistem Informasi, Perguruan Tinggi Raharja
Jl. Jend. Sudirman No. 40 Modern Cikokol Tangerang 15117
Telp. (021) 5529692, 5529586 Faks. (021) 5529742
E-mail: untung.rahardja@faculty.raharja.ac.id, hidayati@faculty.raharja.ac.id

ABSTRAKS

Tujuan diterapkannya sistem informasi oleh berbagai perusahaan secara umum yaitu untuk membantu meningkatkan kinerja dan pelayanan kepada para customer dengan lebih efektif dan efisien. Di dunia Perguruan Tinggi misalnya, guna menghindari kesalahan dan redundancy data, sistem informasi digunakan salah satunya untuk membentuk Jadwal Rencana Study (JRS) mahasiswa setiap semester. Tapi ternyata tidak semudah yang diharapkan, karena masalah perihal bentrok ruang dan waktu terjadi dimana-mana. Untuk menanggulangnya, sementara sudah dilakukan percobaan menggunakan konsep permutation combination yang merupakan penerapan dari cabang ilmu statistik dibidang IT. Konsep tersebut cukup membantu menurunkan jumlah bentrok kurang lebih 50% dari jumlah bentrok semula. Namun, konsep tersebut kurang efektif dan tidak sesuai dengan kebutuhan, bahkan tidak dapat mendeteksi lebih dini jika status bentrok bersifat permanent. Saat ini telah dikembangkan metode baru yang lebih efektif dan dipastikan secara maksimal dapat membantu mengatasi masalah bentrok. Metode tersebut dikenal dengan nama Auto Generated Probabilistic Combination (AGPC). AGPC merupakan metode yang mengkombinasikan urutan berbeda dari pengaturan objek-objek yang penyusunannya dilakukan tanpa pengulangan objek tersebut disetiap urutannya, serta pengurutannya pun dilakukan berdasarkan kriteria status bentrok, dan dipermutasi kembali untuk setiap item per-objek bentroknnya. Dalam artikel ini pun diidentifikasi setidaknya ada 4 masalah yang mendasar perihal konsep permutation combination, mendefinisikan metode Auto Generated Probabilistic Combination (AGPC) sebagai cara baru menanggulangi bentrok, dan terakhir yaitu mengimplementasikan AGPC pada SIS-OJRS di Perguruan Tinggi Raharja. Metode AGPC ini sangat penting untuk dikembangkan khususnya pada proses penyusunan JRS, karena fungsinya yang efektif dalam memberikan warning terhadap status bentrok yang sifatnya permanent sehingga senantiasa menghemat kapasitas hardisk untuk tidak perlu lagi melakukan proses permutasi ulang. Selain itu, proses permutasi yang dilakukan AGPC ini pun sangat sesuai dengan kebutuhan. Bahkan dapat dikatakan bahwa metode AGPC ini dapat mengikis pekerjaan pihak operasional kira-kira 80% dari pekerjaannya semula untuk tidak perlu lagi melakukan proses batal tambah seluruh jadwal bentrok secara manual.

Kata kunci— JRS, Bentrok, Permutation Combination, AGPC.

1. PENDAHULUAN

Penyusunan Jadwal Rencana Study (JRS) dijadikan sebagai salah satu agenda rutin yang dijalankan setiap semester di lingkungan Perguruan Tinggi. Penyusunan tersebut merupakan kegiatan yang dilakukan seluruh mahasiswa aktif untuk mengambil atau menentukan matakuliah yang diambil pada setiap semester. Prosesnya yang rumit dalam hal penentuan jadwal baik itu penentuan jadwal mahasiswa maupun jadwal dosen cukup memakan waktu dan menjadikannya sebagai konsentrasi kerja yang sangat dominan bagi seluruh pribadi saat itu.

Demi menghindari kesalahan dan redundancy data, maka pihak Perguruan Tinggi menentukan sebuah kebijakan untuk menggunakan sebuah sistem informasi untuk melakukan pengolahan proses pembuatan Jadwal Rencana Study (JRS). Tetapi tidak berjalan cukup lancar, karena saratnya data mengakibatkan sistem tersebut menghadapi masalah yang cukup rumit yaitu masalah perihal adanya

bentrok. Awalnya masalah tersebut ditangani secara manual melalui proses batal tambah, tetapi ternyata lagi-lagi waktu tidak memungkinkan untuk kembali melakukan hal-hal yang sifatnya manual.

Ada sebuah konsep yang saat itu dipandang dapat mengatasi masalah bentrok ini. Konsep tersebut dikenal dengan nama *Permutation Combination*. Konsep tersebut dikembangkan dengan mengadopsi prinsip ilmu statistik tentang permutasi dan kombinasi. Cara kerjanya yaitu menyusun kembali urutan mata kuliah yang harus diambil oleh seorang mahasiswa untuk kemudian ditentukan jadwal kelasnya berdasarkan urutan tersebut. Sebagai contoh, misalnya seorang mahasiswa mengambil 4 mata kuliah yaitu mata kuliah A, B, C, dan D. Jika diurutkan untuk pertama kali yaitu menjadi ABCD, maka mata kuliah A adalah sebagai penentu bentrok. Sebagai urutan pertama, mata kuliah A bebas menentukan jadwal dimanapun, berbeda dengan urutan selanjutnya seperti mata kuliah B, C, dan D yang terikat dengan jadwal mata kuliah pada urutan sebelumnya. Mata kuliah B misalnya, harus mencari

jadwal yang tidak bentrok dengan mata kuliah A, dan begitu seterusnya untuk mata kuliah C dan D. Tetapi urutan tersebut tetap akan berstatus bentrok jika ternyata kelas yang dibuka untuk mata kuliah B hanya satu-satunya dan kebetulan bentrok dengan jadwal mata kuliah A yang sudah diambil. Intinya urutan pertama mempunyai prioritas yang utama.

A. Jadwal Rencana Study (JRS)

Menurut data administrasi bagian PSMA Universitas Gunadarma, pengisian Kartu Rencana Study adalah kegiatan yang dilakukan oleh seluruh mahasiswa yang aktif untuk mengambil atau menentukan matakuliah yang diambil pada setiap semester.

Sedangkan menurut naskah SOP Institute Pertanian Bogor (IPB), Penyusunan rencana studi mahasiswa adalah proses penentuan kegiatan pendidikan (pengambilan Mata Ajaran) yang akan dilaksanakan mahasiswa pada semester yang akan datang (termasuk alih semester). Mata Ajaran meliputi, Mata Kuliah, Praktikum, Kolokium/Seminar, Praktek Lapangan, KKN, Magang Kerja dan atau Tugas Akhir.

- KRS adalah Kartu Rencana Studi, berisi daftar Mata Ajaran yang akan diambil mahasiswa pada semester yang akan datang (termasuk alih semester).
- KSM adalah Kartu Studi Mahasiswa, berisi daftar Mata Ajaran yang diambil mahasiswa pada semester berjalan, dibuat berdasarkan KRS.
- Pada naskah SOP UIN Sunan Kalijaga ada tambahan mengenai KPRS. KPRS adalah Kartu Perubahan rencana studi, berisi perubahan mata ajaran yang telah dipilih mahasiswa dalam KRS.

B. Permutasi

Menurut Santoso, Permutasi merupakan penyusunan obyek-obyek yang ada ke dalam suatu urutan tertentu. Hal yang perlu diperhatikan dalam permutasi adalah bahwa obyek-obyek yang ada harus dapat "dibedakan" antara yang satu dengan yang lain.

Menurut Wikipedia bahasa Indonesia, ensiklopedia bebas, Permutasi adalah penyusunan kembali suatu kumpulan objek dalam urutan yang berbeda dari urutan yang semula. Sebagai contoh, kata-kata dalam kalimat sebelumnya dapat disusun kembali sebagai "*adalah Permutasi suatu urutan yang berbeda urutan yang kumpulan semula objek penyusunan kembali dalam dari.*" Sedangkan proses mengembalikan objek-objek tersebut pada urutan yang baku (sesuai ketentuan) disebut sorting.

Pengertian

Jika terdapat suatu untai abjad *abcd*, maka untai itu dapat dituliskan kembali dengan urutan yang berbeda: *acbd*, *dacb*, dan seterusnya. Selengkapnya ada 24 cara menuliskan keempat huruf tersebut dalam urutan yang berbeda satu sama lain.

*abcd abdc acbd acdb adbc adcb
bacd badc bcad bcda bdac bdca
cabd cadb cbad cbda cdab cdba
dabc dacb dbac dbca dcab dcba*

Setiap untai baru yang tertulis mengandung unsur-unsur yang sama dengan untai semula *abcd*, hanya saja ditulis dengan urutan yang berbeda. Maka setiap untai baru yang memiliki urutan berbeda dari untai semula ini disebut dengan permutasi dari *abcd*.

1) Menghitung Banyaknya Permutasi yang Mungkin

Untuk membuat permutasi dari *abcd*, dapat diandaikan bahwa terdapat empat kartu bertuliskan masing-masing huruf, yang hendak kita susun kembali. Juga terdapat 4 kotak kosong yang hendak kita isi dengan masing-masing kartu:

<i>Kartu</i>	<i>Kotak kosong</i>
-----	-----
a b c d	[] [] [] []

Maka kita dapat mengisi setiap kotak dengan kartu. Tentunya setiap kartu yang telah dipakai tidak dapat dipakai di dua tempat sekaligus. Prosesnya digambarkan sebagai berikut:

Di kotak pertama, kita memiliki 4 pilihan kartu untuk dimasukkan.

<i>Kartu</i>	<i>Kotak</i>
-----	-----
a b c d	[] [] [] []
	^ 4 pilihan: a, b, c, d

Sekarang, kondisi kartunya tinggal 3, maka kita tinggal memiliki 3 pilihan kartu untuk dimasukkan di kotak kedua.

<i>Kartu</i>	<i>Kotak</i>
-----	-----
a * c d	[b] [] [] []
	^ 3 pilihan: a, c, d

Karena dua kartu telah dipakai, maka untuk kotak ketiga, kita tinggal memiliki dua pilihan.

<i>Kartu</i>	<i>Kotak</i>
-----	-----
a * c *	[b] [d] [] []
	^ 2 pilihan: a, c

Kotak terakhir, kita hanya memiliki sebuah pilihan.

Kartu	Kotak
-----	-----
a * * *	[b] [d] [c] []
	^ 1 pilihan: a

Kondisi terakhir semua kotak sudah terisi.

Kartu	Kotak
-----	-----
* * * *	[b] [d] [c] [a]

Di setiap langkah, kita memiliki sejumlah pilihan yang semakin berkurang. Maka banyaknya semua kemungkinan permutasi adalah $4 \times 3 \times 2 \times 1 = 24$ buah. Jika banyaknya kartu 5, dengan cara yang sama dapat diperoleh ada $5 \times 4 \times 3 \times 2 \times 1 = 120$ kemungkinan. Maka jika digeneralisasikan, banyaknya permutasi dari n unsur adalah sebanyak $n!$.

2. PERMASALAHAN

Sepintas mungkin metode permutasi combination ini bisa mengatasi bentrok, bahkan telah berhasil menurunkan jumlah bentrok kurang lebih 50% dari jumlah semula. Tetapi ternyata setelah ditelusuri hasilnya tidak maksimal, tidak efektif dan tidak sesuai dengan kebutuhan.

Metode permutasi combination mempunyai cara kerja yaitu menyusun objek-objek kedalam urutan tertentu yang tidak sama dengan urutan sebelumnya. Jadi, metode ini hanya menyusun ulang mata kuliah, tanpa melihat jumlah kelas yang dibuka untuk setiap mata kuliah. Intinya jika ada 2 mata kuliah, sementara kedua mata kuliah tersebut hanya dibuka 1 kelas, dan kebetulan jadwalnya bentrok, maka walaupun dilakukan permutasi berapa kalipun hasilnya pasti bentrok. Inilah sebabnya mengapa metode permutasi combination itu dikatakan tidak efektif. Karena tidak dapat mendeteksi lebih dini bahwa ternyata status bentrok tersebut bersifat permanent.

Bukan hanya itu, metode permutasi combination pun bisa dikatakan tidak sesuai dengan kebutuhan dikarenakan urutan mata kuliah yang dihasilkan tidak sesuai dengan yang seharusnya. Hal ini maksudnya, jika ada seorang mahasiswa mengambil 4 mata kuliah ABCD, dan kebetulan bentrok di 2 mata kuliah yaitu mata kuliah C dan D. Kemudian dilakukan permutasi sehingga urutannya menjadi BACD, apakah mungkin status bentroknnya akan hilang? Jawabannya, kemungkinan besar status bentrok tidak akan hilang. Karena sudah jelas yang bentrok itu adalah mata kuliah C dan D, lalu mengapa permutasinya diawali dari mata kuliah B, padahal sudah jelas mata kuliah B itu tidak bentrok, sehingga tidak perlu dirubah jadwalnya. Nah itu masalahnya mengapa metode permutasi combination ini dikatakan tidak sesuai dengan kebutuhan, karena metode ini kurang sensitif dengan status bentrok.

Berdasarkan hal tersebut dikemukakan ada 4 masalah yang melatarbelakangi pembuatan artikel ini yaitu:

1. Metode seperti apakah yang dapat secara efektif digunakan untuk mengatasi bentrok?
2. Metode seperti apakah yang dapat memberitahukan lebih dini bahwa status bentrok tersebut bersifat permanent?
3. Metode seperti apakah yang dapat secara sensitif membaca status bentrok dan melakukan permutasi untuk mata kuliah yang bentrok tersebut?
4. Metode seperti apakah yang bisa melakukan permutasi bukan hanya berhenti sampai dengan mengurutkan mata kuliah, tetapi juga mengurutkan berdasarkan kelas yang dibuka per-mata kuliah?

3. PEMECAHAN MASALAH

Untuk mengatasi permasalahan seperti yang telah dijelaskan diatas, dapat dilakukan melalui penerapan metode *Auto Generated Probabilistic Combination (AGPC)*. Berikut merupakan 5 ciri khas dari *Auto Generated Probabilistic Combination (AGPC)* yang diterapkan pada proses penanganan bentrok Jadwal Rencana Study (JRS):

1. Diperlukan data kelas-kelas untuk setiap mata kuliah yang berstatus bentrok.
2. Adanya warning jika ternyata seluruh mata kuliah yang berstatus bentrok tersebut hanya dibuka satu kelas, dan permutasi tidak dilakukan.
3. Permutasi dilakukan untuk seluruh kelas bagi setiap mata kuliah yang berstatus bentrok, dan diikuti dengan pencarian kelas-kelas bagi mata kuliah lainnya yang berstatus tidak bentrok.
4. Jika dihasilkan jadwal yang tidak bentrok, maka permutasi dihentikan.
5. Jika sampai dengan proses permutasi selesai dilakukan ternyata seluruh jadwal yang dihasilkan masih berstatus bentrok, maka seluruh jadwal tersebut ditampilkan untuk dapat dilakukan pemilihan baik dijadikan sebagai jadwal tetap menggantikan jadwal yang lama, maupun dilakukan proses permutasi ulang berdasarkan salah satu jadwal tersebut.

Untuk dapat menangani masalah bentrok secara efektif, diperlukan suatu pemeriksaan awal untuk menilai apakah status bentrok tersebut merupakan bentrok yang sifatnya permanent ataukah bukan? Jika iya, maka perlu adanya warning yang menyatakan bahwa status bentrok tidak bisa dihilangkan, sehingga proses permutasi tidak perlu dilakukan. Hal ini berkaitan dengan point no [1] dan [2].

Bukan hanya itu, cara kerja AGPC pun berbeda dengan permutasi combination pada umumnya. AGPC mendahulukan proses permutasi untuk mata kuliah berstatus bentrok terlebih dahulu dengan tujuan mengatasi masalah pada tempat yang

bermasalah. Hal ini pula yang menjadikan AGPC menjadi lebih efektif.

Proses permutasi yang dilakukan AGPC bukan lagi permutasi berdasarkan mata kuliah, melainkan merupakan permutasi yang dilakukan berdasarkan kelas-kelas pada setiap mata kuliah. Hal ini berkaitan dengan point no [3]. Berikut adalah contohnya :

Jika ada sebuah jadwal berstatus bentrok. Jadwal itu terdiri dari 5 mata kuliah yaitu terdiri dari mata kuliah A, B, C, D, dan E. Serta ada 3 mata kuliah yang berstatus bentrok yaitu mata kuliah A, B, dan D. Setelah diperiksa, ternyata mata kuliah A dibuka 2 kelas yaitu A1 dan A2. Mata kuliah B dibuka 1 kelas yaitu B1. Sementara mata kuliah C dibuka 2 kelas yaitu C1 dan C2. Maka jumlah proses permutasi bagi ketiga mata kuliah tersebut yaitu didapat dari rumus berikut :

$$P = \text{jumlah MK A} \times \text{jumlah MK B} \times \text{jumlah MK C} \\ = 2 \times 1 \times 2 = 4$$

Susunannya yaitu :

1. A1, B1, C1
2. A1, B1, C2
3. A2, B1, C1
4. A2, B1, C2

Kemudian dilanjutkan dengan penentuan kelas-kelas bagi mata kuliah lainnya yang berstatus tidak bentrok. Hal ini tentunya dengan menyesuaikan pada jadwal-jadwal mata kuliah sebelumnya. Ketika dihasilkan sebuah susunan jadwal yang tidak bentrok, meskipun permutasi belum selesai, maka proses permutasi dihentikan. Kemudian sistem memberikan pilihan apakah jadwal tersebut ingin dijadikan jadwal tetap menggantikan jadwal yang sebelumnya ? Hal ini merupakan penjelasan dari point no [4].

Lain halnya jika sampai dengan proses permutasi selesai, seluruh jadwal yang dihasilkan berstatus bentrok semua. Hal ini berkaitan dengan point no [5]. Maka sistem akan menampilkan seluruh jadwal yang telah dibuat. Dan user diberi pilihan, apakah ingin memilih salah satu dari jadwal-jadwal tersebut, ataupun ingin melakukan proses permutasi ulang dari jadwal-jadwal tersebut.

4. IMPLEMENTASI

Penanganan bentrok menggunakan metode *Auto Generated Probabilistic Combination (AGPC)* sudah diimplementasikan pada Perguruan Tinggi Raharja, yakni pada sistem informasi SIS OJRS (Online JRS). Students Information Services, atau yang biasa disingkat SIS, merupakan sistem yang dikembangkan oleh Perguruan Tinggi Raharja untuk tujuan sebagai sistem pelayanan informasi kepada mahasiswa secara optimal. Pengembangan SIS juga merupakan akses publikasi bagi Perguruan Tinggi Raharja di bidang ilmu komputer dan dunia IT khususnya.

SIS sudah dikembangkan ke dalam beberapa versi, dimana masing-masing merupakan kelanjutan

dari SIS versi sebelumnya. SIS OJRS (Online Jadwal Rencana Studi) merupakan SIS versi ke-4. Sesuai namanya, SIS OJRS dibuat untuk kebutuhan perkuliahan mahasiswa, yaitu untuk menyiapkan JRS (Jadwal Rencana Studi).

Hasil akhir yang ingin dicapai dari SIS OJRS ini yaitu dihasilkanlah JRS (Jadwal Rencana Studi) dan KST (Kartu Studi Tetap) yang jumlah bentroknnya seminimal mungkin bahkan hingga mencapai 0%. Karena itulah, untuk dapat mereduksi bentrok dengan cara yang efektif, diterapkanlah metode *Auto Generated Probabilistic Combination (AGPC)* ini.

Algoritma 1	Algoritma 2
:: Repair :: Auto Repair :: Auto List ::	:: Auto List Repair ::

STMIK RAHARJA DAN AMIK RAHARJA INFORMATIKA
 Kampus Modern : Jl. Jendral Sudirman No. 40 Babakan Cikokol-Tangerang 15117 Telp. 5529692
 Kampus Cimone : Jl. Gatot Subroto Km. 2 Harmoni Mas A7-B dan A18-19 Cimone Tangerang 15114
 Telp. 5537809 Fax. 5532019

Nomor Induk Mahasiswa : 0833361793 Jurusan : Teknik Informatika
 Nama Mahasiswa : Muchlis Saputra Konsentrasi : Artificial Informatics
 Jenjang Studi : Diploma 3

KARTU STUDI TETAP

No	Kode_Kelas	Sks	Hari	Ruang	Jam Mulai	Jam Akhir	Ket
1	MT183E	2	Jumat	M-104	9:40:00 AM	11:20:00 AM	OK
2	MT112A	2	Rabu	M-201	9:40:00 AM	11:20:00 AM	OK
3	MT103A	2	Rabu	M-104	12:10:00 PM	1:50:00 PM	Bentrok
4	PR183B	3	Rabu	M-206	12:10:00 PM	1:50:00 PM	Bentrok
5	PR201I	3	Sabtu	M-201	8:00:00 AM	9:40:00 AM	OK
6	SI103I	3	Selasa	M-206	1:00:00 PM	2:40:00 PM	OK

Gambar 1. Kartu Studi Tetap (KST) pada SIS OJRS

Gambar di atas merupakan tampilan KST mahasiswa. Dari halaman tersebut dapat diketahui informasi mengenai kelas apa saja didapat oleh seorang mahasiswa semester ini, hari apa, ruangnya dimana, kapan waktunya, serta apa status kelasnya, yaitu bentrok ataukah tidak kelasnya. Pada bagian atas KST terdapat juga sebuah link teks "Auto_List_Repair". Fungsinya yaitu untuk menjalankan metode AGPC ini.

Jika ternyata kedua mata kuliah yang tertera berstatus bentrok pada KST diatas yaitu mata kuliah MT103 dan PR183 masing-masing hanya dibuka 1 kelas, maka akan tampil sebuah warning yang menyatakan bahwa status bentrok bersifat permanent, sehingga status bentrok tidak dapat dihilangkan dan proses permutasi tidak dijalankan. Berikut tampilan warningnya.

BENTROK TIDAK DAPAT DIHILANGKAN

<<-- [Back To KST](#)

Gambar 2. Warning pada proses AGPC

Tetapi lain halnya jika salah satu dari kedua mata kuliah tersebut dibuka lebih dari 1 kelas, maka permutasi tetap dijalankan. Namun apabila pada saat proses permutasi kemudian dihasilkan susunan jadwal yang berstatus tidak bentrok, maka proses permutasi dihentikan. Kemudian ditampilkan data susunan jadwal hasil proses permutasi tersebut. Berikut tampilannya.

Tampil Log Perpindahan KST Mahasiswa

Jumlah Urutan : 2

NIM : 0833361793

DITEMUKAN KELAS TIDAK BENTROK

Ke	No	NIM	Kode_MK	Kelas	Jml_Kelas	Kode_Waktu	Kode_Waktu2	Bentrok	Ket
1	5	0833361793	PA102	3	4	301	302	0	Ulang
	6	0833361793	FR201	7	8	505	506	0	
	7	0833361793	SI103	7	5	201	202	0	
	8	0833361793	UM100	2	2	110	111	0	
2	1	0833361793	FR183	2	1	306	307	0	
	2	0833361793	MT103	2	2	503	504	0	
	3	0833361793	MT112	7	4	208	209	0	
	4	0833361793	MT183	4	4	501	502	0	
	5	0833361793	PA102	3	4	301	302	0	
	6	0833361793	FR201	7	8	505	506	0	
	7	0833361793	SI103	7	5	201	202	0	
	8	0833361793	UM100	2	2	110	111	0	

Gambar 3. Tampilan Log AGPC jika ditemukan susunan jadwal yang tidak bentrok

Pada gambar tersebut hanya dihasilkan 2 proses permutasi, dimana permutasi yang ke-2 tersebut berisi susunan jadwal yang tidak bentrok. Ketika di klik pada tombol OK, berarti jadwal tersebut telah disetujui untuk menggantikan jadwal lama pada KST mahasiswa yang bersangkutan. Berikut ditampilkan KST baru setelah tombol OK dieksekusi.

STMIK RAHARJA DAN AMIK RAHARJA INFORMATIKA
 Kampus Modern : Jl. Jendral Sudirman No. 40 Babakan Cikokol-Tangerang 15117 Telp. 55296927
 Kampus Cimone : Jl. Gatot Subroto Km. 2 Harmoni Mas A7-8 dan A18-19 Cimone Tangerang 15114
 Telp. 5537809 Fax. 5532019

Nomor Induk Mahasiswa : 0833361793 Jurusan : Teknik Informatika
 Nama Mahasiswa : Muchlis Saputra Konsentrasi : Artificial Intelligence
 Jenjang Studi : Diploma 3

KARTU STUDI TETAP

No	Kode_Kelas	Sks	Hari	Ruang	Jam Mulai	Jam Akhir	Ket
1	MT183D	2	Jumat	M-104	8:00:00 AM	9:40:00 AM	OK
2	MT103B	2	Jumat	M-103	9:40:00 AM	11:20:00 AM	OK
3	PR201G	3	Jumat	M-201	11:20:00 AM	1:00:00 PM	OK
4	PA102C	3	Rabu	M-201	8:00:00 AM	9:40:00 AM	OK
5	PR183B	3	Rabu	M-206	12:10:00 PM	1:50:00 PM	OK
6	SI103G	3	Selasa	M-206	8:00:00 AM	9:40:00 AM	OK
7	MT112G	2	Selasa	M-104	1:50:00 PM	3:30:00 PM	OK
8	UM100B	2	Senin	M-202	3:30:00 PM	5:10:00 PM	OK

Jumlah Satuan Kredit Semester (SKS) 20

Gambar 4. Kartu Studi Tetap (KST) pada SIS OJRS hasil AGPC

Ada satu kondisi lagi pada metode AGPC ini yaitu apabila pada saat proses permutasi selesai dijalankan, namun susunan jadwal yang dihasilkan seluruhnya tetap berstatus bentrok, maka jadwal-

jadwal tersebut juga akan ditampilkan. Berikut tampilannya.

Tampil Log Perpindahan KST Mahasiswa

Jumlah Urutan : 6

NIM : 0912462438

BENTROK SEMUA

Ke	No	NIM	Kode_MK	Kelas	Jml_Kelas	Kode_Waktu	Kode_Waktu2	Bentrok	Ket
1	1	0912462438	SI152	7	2	514	515	1	
	2	0912462438	SI171	3	3	514	515	1	
	3	0912462438	MT103	3	3	512	513	0	
	4	0912462438	MT112	3	4	612	613	1	Bentrok (Ulang)
	5	0912462438	PA102	4	4	614	615	0	
	6	0912462438	FR201	10	8	312	313	0	
	7	0912462438	SI103	5	5	612	613	1	
	8	0912462438	SI330	4	2	214	215	0	
2	1	0912462438	SI152	7	2	514	515	0	
	2	0912462438	SI171	4	3	612	613	1	
	3	0912462438	MT103	3	3	512	513	0	
	4	0912462438	MT112	2	4	610	611	0	Bentrok (Ulang)

Gambar 5. Tampilan Log AGPC jika tidak ditemukan susunan jadwal yang tidak bentrok

Pada kolom "Ket" yang tertera pada gambar [5] diatas, tertulis bahwa semua jadwal yang tersusun berstatus "Bentrok". Apabila status Bentrok tersebut diklik, berarti jadwal berstatus bentrok tersebut telah disetujui untuk menggantikan jadwal lama pada KST mahasiswa yang bersangkutan. Namun apabila ingin mengulangi metode AGPC berdasarkan status bentrok pada salah satu jadwal yg telah tersusun tersebut, cukup dengan klik pada tombol "Ulangi" yang tertera dibawah status "Bentrok". Maka prose permutasi dieksekusi kembali.

Database

SIS OJRS yang diimplementasikan pada Perguruan Tinggi Raharja menggunakan database SQL Server. Di dalam database server tersebut terintegrasi berbagai database baik itu database yang digunakan oleh Student Information Services (SIS), maupun Green Orchestra (GO), Raharja Multimedia Edutainment (RME), dll. SIS OJRS menggunakan database yang sama seperti database yang digunakan oleh Student Information Services (SIS).

Pada database inilah dibuat tabel-tabel yang dibutuhkan berkenaan dengan proses AGPC. Terdapat dua macam tabel yang harus disiapkan, yaitu: tabel CT_List yang berisi data permutasi kelas-kelas bagi mata kuliah yang berstatus bentrok, dan tabel CT_List2 merupakan tabel gabungan yang berisi data permutasi kelas-kelas bagi mata kuliah berstatus bentrok dan berstatus tidak bentrok.

Column Name	Data Type	Length	Allow Nulls
Ke	int	4	✓
Kode_MK	varchar	10	✓
NIM	varchar	10	✓
[No]	int	4	✓
Kelas	int	4	✓
Jml_Kelas	int	4	✓

Gambar 6. Struktur tabel Tbl CT_List

Tabel di atas merupakan tabel utama yang merupakan tempat penyimpanan data yang diperlukan untuk melakukan penyimpanan data awal proses permutasi. *Field-field* yang dibutuhkan disesuaikan dengan sistem yang ada. *Field* Ke, Kode_MK, NIM, No, Kelas dan Jml_Kelas merupakan *field* yang menjelaskan data yang harus terpenuhi sebelum proses permutasi benar-benar dijalankan. Isi daripada *field* Jml_Kelas yang menentukan apakah proses permutasi layak untuk dilakukan ataukah tidak. Jika layak, maka proses permutasi mulai dieksekusi dan proses tersebut dimasukkan kedalam tabel CT_List2.

Column Name	Data Type	Length	Allow Nulls
Ke	int	4	✓
Kode_MK	varchar	10	✓
NIM	varchar	10	✓
[No]	int	4	✓
Jml_Kelas	int	4	✓
Kelas	int	4	✓
Kode_Waktu	varchar	5	✓
Kode_Waktu2	varchar	5	✓
Bentrok	int	4	✓

Gambar 7. Struktur tabel Tbl CT_List2

5. KESIMPULAN

Auto Generated Probabilistic Combination (AGPC) merupakan salah satu bagian penting pada proses pembuatan Jadwal Rencana Study (JRS). Cara kerjanya yang sensitif melakukan permutasi berdasarkan status bentrok membuat AGPC ini secara tepat menangani bentrok tepat ke inti masalahnya. Selain itu AGPC pun dapat memberikan warning awal jika status bentrok tersebut bersifat permanent, sehingga proses permutasi tidak perlu dilakukan. Hal ini sangat membantu karena selain menghemat kapasitas hardisk, AGPC pun memberikan status yang jelas mengapa jadwal tersebut tidak dapat hilang status bentroknnya.

PUSTAKA

- [1] Andi (2005). *Aplikasi Web Database ASP Menggunakan Dreamweaver MX 2004*. Yogyakarta: Andi Offset.
- [2] Bernard, R, Suteja (2006). *Membuat Aplikasi Web Interaktif Dengan ASP*. Bandung: Informatika.
- [3] Untung Rahardja (2007). *Pengembangan Students Information Services di Lingkungan Perguruan Tinggi Raharja*. Laporan Pertanggung Jawaban. Tangerang: Perguruan Tinggi Raharja.
- [4] Anonim (2009). Standar Operating Procedure (SOP) Institute Pertanian Bogor. Diakses pada 5

Mei 2009 dari :
<http://72.14.235.132/search?q=cache:o2fybi0up0sJ:bima.ipb.ac.id/mLLIS%2520ajmp/Naskah-SOP-AJMP.doc+Definisi+Jadwal+Rencana+Study&cd=2&hl=id&ct=clnk&gl=id>

- [5] Anonim (2009). Standar Operating Procedure (SOP) Penyusunan KRS dan KPRS UIN Sunan Kalijaga. Diakses pada tanggal 5 Mei 2009 dari : http://www.uin-suka.info/joomlakesuka/jamu/SOP_akademik/penyusunan_krs.htm
- [6] Santoso (2009). Materi I : Permutasi dan Kombinasi. Diakses pada tanggal 5 Mei 2009 dari : <http://ssantoso.blogspot.com/2009/03/materi-i-permutasi-dan-kombinasi.html>