

Prototype sistem pakar untuk penjadwalan

Catur Edi Widodo

Departemen Fisika, Fakultas Sains dan Matematika, Universitas Diponegoro, Semarang

E-mail: catur.ediwidodo@gmail.com

ABSTRACT

Scheduling is a complex problem, which requires huge resource to solve. The resources required include faculty, lecturer, classrooms, and the period of time for teaching. The solution of academic scheduling in large scale still face up so many obstacles to done manually. The college has to give a schedule in certain time when every academic activity is not crash. Scheduling is needed to anticipate crash of students hours to study and lectures time to teach. Scheduling have to fill the boundary and condition so that it convenient when it used. Under these conditions, a system is needed to set the schedule will not crash so as to improve the work of everyone. The possibility to find the best result and the implemented method approach to solve the problem is use constrain satisfaction method. Making this scheduling application is started with build table of combination from class data, room data, lecture data and time slot and followed by initiation and calculation with genetic algorithm. During the process, we generate the time and room data of each lecturer and test if crash or no crash. From the result, indicate fine schedule means there is no crash between each other and all class can be scheduled. Scheduling is optimal if all of space and time that is provided can be filled without happen crash

Keywords: *scheduling, generate and test, constraint satisfaction, crash.*

ABSTRAK

Penjadwalan merupakan masalah yang kompleks, yang membutuhkan sumber daya yang besar untuk memecahkannya. Sumber daya yang diperlukan termasuk fakultas, dosen, ruang kelas, dan jangka waktu untuk mengajar. Solusi penjadwalan akademik dalam skala besar masih menghadapi banyak hambatan untuk dilakukan secara manual. Perguruan tinggi harus memberikan jadwal waktu tertentu sehingga setiap kegiatan akademik tidak tabrakan. Penjadwalan diperlukan untuk mengantisipasi tabrakan jam untuk matakuliah dan tabrakan jam dosen dalam memberikan beberapa mata kuliah. Penjadwalan harus mengisi batas dan kondisi sehingga nyaman ketika digunakan. Dengan kondisi tersebut, maka dibutuhkan suatu sistem untuk mengatur jadwal agar tidak terjadi tabrakan sehingga dapat meningkatkan kerja semua orang. Kemungkinan untuk menemukan hasil terbaik dan metode pendekatan yang dapat diterapkan untuk memecahkan masalah adalah penggunaan metode constrain stisfaction. Pembuatan aplikasi penjadwalan ini dimulai dengan membangun tabel kombinasi dari data kelas, data ruang, data kuliah dan slot waktu dan diikuti oleh inisiasi dan perhitungan dengan algoritma Generate and Test. Selama proses tersebut, dibangkitkan waktu dan ruang kelas masing-masing dosen dan dilakukan test apakah terjadi tabrakan. Dari hasil penelitian, jadwal dapat disusun tanpa terjadi tabrakan antara satu dengan yang lain dan semua kelas dapat dijadwalkan. Penjadwalan optimal jika semua ruang dan waktu yang disediakan dapat diisi tanpa terjadi tabrakan

Kata kunci: *penjadwalan, generate and test, constraint satisfaction, tabrakan.*

PENDAHULUAN

Penjadwalan merupakan pekerjaan rutin yang dilakukan setiap semester dalam sistem akademik perguruan tinggi. Ada dua penjadwalan yang sering dijumpai pada perguruan tinggi yaitu penjadwalan perkuliahan dan ujian, baik teori maupun praktikum [1]. Hingga saat ini penjadwalan

mata kuliah pada institusi pendidikan merupakan masalah yang sangat kompleks. Inti dari penjadwalan mata kuliah adalah bagaimana menjadwalkan sejumlah komponen yang terdiri atas kelas, mata kuliah, dosen, ruang dan waktu dengan sejumlah batasan dan syarat tertentu. Penjadwalan perkuliahan termasuk dalam golongan jenis *timetabling*. Permasalahanan *timetabling* dapat digolongkan

sebagai *NP-Hard Problem* (*nondeterministic polynomial time*). Permasalahan *NP-Hard Problem* merupakan permasalahan yang apabila seluruh kombinasi alternatif dicobakan untuk diuji, maka waktu yang dibutuhkan untuk mencari solusi yang *feasible* dari permasalahan akan meningkat tajam [2]. Oleh karena itu penyelesaian permasalahan optimasi akan sulit dilakukan dengan menggunakan metode optimasi konvensional (Kanoh dan Sakamoto, 2008). Proses penjadwalan adalah suatu proses untuk menerapkan *event* yang berisi komponen mata kuliah, dosen, kelas dan semester pada *time slot* yang berisi komponen waktu dan ruang [3]. Jika menggunakan sistem manual maka masalah ini membutuhkan waktu proses yang cukup lama untuk pencarian solusinya, terlebih lagi bila ukuran permasalahan semakin besar dengan bertambahnya jumlah komponen dan tetapan atau syarat yang ditentukan oleh institusi tempat jadwal tersebut di gunakan.

Pembuatan jadwal akan selalu muncul karena harus dilakukan pada setiap pergantian semester. Umumnya jadwal diselesaikan dengan membuat tabel jadwal secara manual. Cara ini membutuhkan waktu yang lama, karena pembuatan jadwal tersebut sangatlah kompleks yang terdiri dari beberapa komponen penyusun, seperti mata kuliah, dosen, ruang, dan waktu. Dosen juga dimasukkan dalam komponen penyusun karena dosen diwajibkan hadir pada saat proses belajar mengajar berlangsung. Pada setiap komponen penyusun tersebut banyak terdapat aturan dan batasan-batasan yang telah ditentukan, oleh karena itu diperlukan penjadwalan otomatis yang dapat membuat jadwal dengan cepat, mudah dan tetap harus memperhatikan aturan-aturan.

Banyak aspek yang harus dipertimbangkan untuk memproses jadwal kuliah yang optimal. Oleh karena itu perlu ditetapkan suatu batasan yang menjadi acuan dalam proses penyusunan jadwal kuliah. Berbagai aspek yang berkaitan dalam penjadwalan kuliah tersebut dan harus dilibatkan dalam pertimbangan diantaranya :

1. Banyaknya dosen yang mengajar beberapa mata kuliah.
2. Satu mata kuliah bisa mempunyai beberapa kelas
3. Seorang dosen bisa mengampu lebih dari satu mata kuliah di beberapa kelas
4. Tidak boleh adanya jadwal kuliah yang saling bentrok antar dosen, kelas, ruang ataupun waktu perkuliahan.

Sistem penjadwalan kuliah di beberapa institusi perguruan tinggi sampai saat ini masih dilakukan secara manual, yaitu dengan pencarian blok-blok atau kolom-kolom mana saja yang masih kosong, kemudian menempatkan jadwal pada blok atau kolom tersebut. Jadwal yang dihasilkan dengan cara seperti ini memerlukan waktu yang cukup lama dan cenderung mengabaikan berbagai aspek tersebut, sehingga jadwal kuliah yang sudah dibuat seringkali perlu dilakukan perbaikan lagi. Oleh karena itu perlu dikembangkan suatu sistem penjadwalan kuliah yang dapat mengakomodasi berbagai aspek yang menjadi pertimbangan diatas [3].

Sistem penjadwalan secara manual akan membuat bingung dan memerlukan ketelitian yang tinggi, Karena terdapat hal hal lain yang harus diperhatikan antara lain

1. Satu ruangan hanya boleh digunakan oleh satu jadwal kuliah, Misal untuk satu mata kuliah 2 sks maka perlu dialokasikan jam 7.00 sampai dengan jam 10.00, yang berarti untuk jam tersebut disebuah ruang kuliah hanya boleh digunakan oleh satu mata kuliah pada satu kelas.
2. Tidak semua jam kuliah dapat digunakan oleh dosen tertentu, misal dosen pengampu mata kuliah tidak bisa hadir pada hari senin pada jam tertentu atau dosen tersebut tidak bisa hadir setiap hari senin, dosen tersebut hanya bisa hadir selain hari senin
3. Satu dosen tidak mungkin hadir di satu blok jadwal untuk dua mata kuliah yang berbeda atau satu mata kuliah untuk dua kelas dalam waktu bersamaan

Dengan kemajuan ilmu pengetahuan dalam bidang komputasi cerdas, masalah

penjadwalan yang telah disebutkan sebelumnya dapat diotomatisasi sehingga dapat memberikan solusi optimal sesuai dengan batasan dan syarat yang sudah ditentukan. Sejumlah metode dari berbagai disiplin ilmu telah diusulkan dalam literatur, seperti: riset operasi, kecerdasan buatan, dan kecerdasan komputasional. Metode-metode tersebut dapat dibagi ke dalam 4 (empat) kategori, yaitu: *Sequential Methods*, *Cluster Methods*, *Constraint Based Methods*, dan *Meta-heuristic Methods*, seperti *genetic algorithms*, *simulated annealing*, dan *tabu search* [4]. Tujuan dari penelitian ini adalah menerapkan algoritma generate and test untuk penjadwalan kuliah di Universitas dengan sistem SKS.

DASAR TEORI

Penjadwalan adalah penempatan sumber daya (*resource*) dalam satu waktu. Penjadwalan mata kuliah merupakan persoalan penjadwalan umum dan sulit yang tujuannya adalah menjadwalkan pertemuan dari sumber daya. Sumber daya yang dimaksud adalah dosen pengampu mata kuliah, mata kuliah, ruang kuliah, kelas mahasiswa, dan waktu perkuliahan.

Terdapat batasan/persyaratan (*constraints*) dalam penyusunan penjadwalan mata kuliah dan ujian akhir semester. *Constraint* sendiri merupakan suatu syarat tidak boleh terjadi pelanggaran terhadap kendala yang ditetapkan agar dapat menghasilkan susunan penjadwalan yang baik. Beberapa *constraint* tersebut, yaitu:

- a) Dosen tidak boleh dijadwalkan lebih dari satu kali pada waktu yang bersamaan
- b) Satu kelas dan ruang tidak boleh dijadwalkan lebih dari satu kali pada waktu yang bersamaan.

Jika terjadi pelanggaran terhadap kendala yang ditetapkan maka akan diberikan suatu nilai penalti atau hukuman antara 0 sampai 1 untuk setiap pelanggaran. Semakin kecil jumlah pelanggaran yang terjadi solusi

penjadwalan yang dihasilkan akan semakin baik [5].

Penelitian tentang *timetabling* telah banyak dilakukan. *Course timetabling* dapat diselesaikan dengan menggunakan kombinasi *local search* dan algoritma genetika untuk mencari solusi yang optimal dalam memenuhi *soft constraints* [6]. Penjadwalan yang baik didapatkan dari beberapa individu selama proses *running* dalam model. Metode yang dapat digunakan untuk penyelesaian diantaranya adalah *knowledge based* (KB). Metode *knowledge based* akan menghasilkan *partial solution* yang dapat dimasukkan dalam model *Genetic Algorithm* (GA) di setiap generasinya. Model algoritma genetika dengan penambahan informasi penjadwalan periode sebelumnya (*knowledge based*) lebih baik daripada model algoritma genetika tanpa penambahan informasi penjadwalan periode sebelumnya.

Ada lima jenis kelas dalam satu semester yang harus dipenuhi ketentuannya dalam melaksanakan tugas pengajaran agar *weekly timetabling* menjadi *feasible*. Ke-lima kelas itu adalah *lecture*, *tutorial*, *project*, *design*, dan *laboratory*. *Constraints* yang digunakan dalam model, yaitu *Hard*, *medium*, dan *soft constraints*. Salah satu metode yang digunakan untuk mendapatkan nilai optimal adalah *Simulated Annealing* [7].

Optimasi penjadwalan dapat dilakukan dengan tujuan untuk meratakan beban belajar mahasiswa dan mendistribusikan mata ujian secara merata. Metode *Max Min Ant Systems* (MMAS) dapat digunakan untuk menemukan solusi optimal [8]. Metode *integer programming* untuk penjadwalan *Shift Assignment* pada tenaga pengajar, didapatkan penjadwalan pengajaran *feasible* yang tidak mengalami bentrok, dan adanya kesesuaian jadwal antara tenaga pengajar dan mahasiswa [9].

Pembuatan jadwal secara otomatis dapat membantu administrator dalam pembuatan jadwal dalam hal waktu, sehingga siapapun dapat membuat dan mengatur variasi jadwal

pada perguruan tinggi tersebut. Selain itu pembuatan jadwal secara otomatis lebih akurat dan teliti dibandingkan dengan dibuat secara manual [1].

Algoritma Generate and Test dapat digunakan untuk penjadwalan dengan *timeslot*, kapasitas tempat duduk dan ruang yang terbatas. Dapat juga digunakan untuk penjadwalan ujian praktek, pengaturan kapasitas ruang untuk ujian yang mana setiap ruang berbeda kapasitas tempat duduk untuk setiap dosen yang berbeda [10].

Masalah-masalah yang ada dalam penjadwalan ujian dapat di atasi dengan menggunakan metode yang berdasarkan kecerdasan buatan. Generate and Test adalah salah satu metode pencarian heuristik dalam terminology kecerdasan buatan yang cukup dikenal. Generate and Test merupakan algoritma paling sederhana dalam teknik pencarian heuristik. Dalam Generate and Test terdapat dua prosedur penting yaitu generate (membangkitkan) yaitu membangkitkan semua solusi yang mungkin dan test (pengujian) yaitu menguji solusi yang dibangkitkan tersebut apakah memenuhi constraint yang disyaratkan, atau yang harus dipenuhi. Adapaun algoritma Generate and Test adalah sebagai berikut :

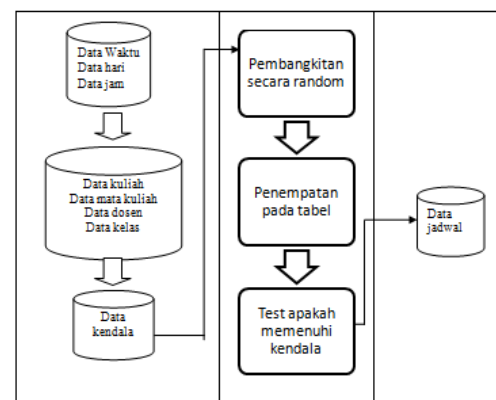
1. Bangkitkan sebuah solusi yang mungkin
2. Uji tiap-tiap node yang merupakan solusi dengan cara membandingkan node tersebut dengan node akhir dari suatu lintasan yang dipilih dengan kumpulan tujuan yang diharapkan.
3. Jika solusi telah ditemukan, maka keluar dari sistem. Jika belum menemukan solusi, maka kembali ke langkah 1.

METODE PENELITIAN

Penelitian ini terbagi ke dalam 3 tahapan yakni tahapan pengumpulan data, tahapan perancangan dan tahapan *coding*. Tahapan pengumpulan data dilakukan dengan wawancara, studi literature dan pembuatan model. Dalam hal ini data yang dibutuhkan

dalam penelitian ini adalah data dosen, data mata kuliah, data ruang, data kelas dan data Jam Mengajar Pada tahapan perancangan dilakukan perancangan aplikasi penjadwalan yang meliputi perancangan *flowchart*, perancangan basis data dan perancangan *user interface* .

Kerangka sistem informasi penjadwalan mata kuliah secara garis besar dibagi menjadi tiga tahapan yaitu input, proses dan output seperti pada Gambar 1. Dalam tahapan input dilakukan masukan data antara lain data waktu, data hari, data jam, data kuliah, data mata kuliah, data dosen, data kelas, data larangan dan data ruang. Dalam tahap proses di implementasikan algoritma generate and test antara lain pembangkitan angka random, evaluasi kendala dan penempatan pada tabel. Pada tahapan output didapatkan hasil jadwal dari tahapan proses.



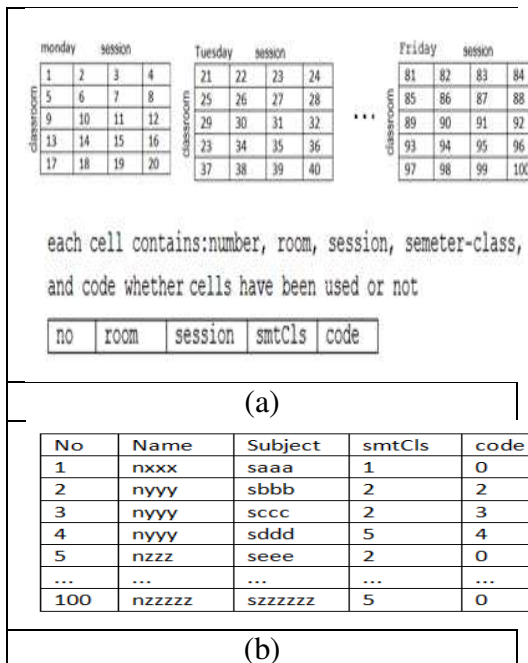
Gambar 1. Kerangka Sistem

Dalam Perancangan Proses algoritma yang menggambarkan proses jalannya proses program yang akan dibuat dari input data, proses data sampai dengan output ataupun informasi yang dihasilkan. Adapun algoritma yang disusun adalah sebagai berikut:

```

filling in the same session with the different classes
filling in the lecturer-lectures tables with the data number, name,
subject, semester-class, and code
(code 0 means the lecturer teaching the subject)
(Code 1 means lecturer teaching Monday and so forth)
run the algorithm generate and test:
  place the randomly numbers of lecturer-lectures
  in room-session table, then check:
    in == 0 and
    smtCls of lecturer-lectures. == smtCls of room-session.
    if yes then code == 1
    if not then repeat generate and test
    
```

Setelah tahap penyusunan algoritma selesai, tahapan dilanjutkan dengan tahap perancangan basis data sebagai tempat penyimpanan data yang akan digunakan dalam proses pembuatan jadwal. Rancangan basis data di gambarkan dalam bentuk tabel relasi yaitu room-session table dan lectures-lecturer table seperti Gambar 2.



Gambar 2. Rancangan Basis Data.
(a)Tabel Room–Session (b) Tabel Lectures–Lecturer

HASIL DAN PEMBAHASAN

Dari koding yang telah dilakukan berdasarkan algoritma yang disusun, jika dieksekusi akan menghasilkan pasangan antara nama dosen – matakuliah dan ruang kelas. Dalam laporan ini di tampilkan dua buah hasil eksekusi (Gambar 3) dimana keduanya optimal dalam arti tidak terjadi tabrakan sama sekali.

[97 35] [98 60] [99 78] [1 62] [2 3] [3 58] [4 99]
[5 77] [6 81] [7 89] [8 33] [9 6] [10 41] [11 84]
[12 37] [13 26] [14 48] [15 44] [16 36] [17 52] [18 28]
[19 7] [20 65] [21 14] [22 55] [23 70] [24 95] [25 69]
[26 13] [27 17] [28 21] [29 79] [30 23] [31 67] [32 29]
[33 57] [34 31] [35 100] [36 63] [37 50] [38 43] [39 40]
[40 72] [41 19] [42 88] [43 83] [44 27] [45 46] [46 30]
[47 73] [48 92] [49 96] [50 11] [51 49] [52 54] [53 24]
[54 12] [55 66] [56 74] [57 87] [58 85] [59 15] [60 2]
[61 25] [62 32] [63 80] [64 45] [65 42] [66 39] [67 61]
[68 8] [69 34] [70 5] [71 56] [72 71] [73 93] [74 76]
[75 16] [76 4] [77 20] [78 94] [79 68] [80 97] [81 38]
[82 47] [83 9] [84 75] [85 98] [86 1] [87 53] [88 59]
[89 91] [90 18] [91 86] [92 64] [93 90] [94 51] [95 82]
[96 10] [100 22]

(a)

[97 22] [98 60] [99 66] [1 4] [2 38] [3 25] [4 17]
[5 23] [6 19] [7 76] [8 35] [9 18] [10 7] [11 24]
[12 67] [13 58] [14 77] [15 3] [16 69] [17 5] [18 47]
[19 54] [20 28] [21 99] [22 9] [23 75] [24 50] [25 40]
[26 88] [27 63] [28 42] [29 39] [30 96] [31 27] [32 2]
[33 20] [34 31] [35 10] [36 26] [37 80] [38 8] [39 74]
[40 16] [41 43] [42 95] [43 34] [44 85] [45 13] [46 91]
[47 98] [48 68] [49 48] [50 93] [51 87] [52 52] [53 46]
[54 61] [55 73] [56 86] [57 65] [58 97] [59 36] [60 90]
[61 78] [62 89] [63 30] [64 70] [65 6] [66 1] [67 51]
[68 72] [69 71] [70 79] [71 32] [72 44] [73 41] [74 21]
[75 100] [76 14] [77 45] [78 94] [79 11] [80 33] [81 12]
[82 59] [83 55] [84 62] [85 57] [86 81] [87 49] [88 29]
[89 56] [90 82] [91 15] [92 64] [93 37] [94 92] [95 83]
[96 53] [100 84]

(b)

Gambar 3. Hasil Eksekusi Program
(a) Eksekusi I (b) Eksekusi II

Dari hasil eksekusi I terlihat angka 97 dan 35 artinya dosen-matakuliah nomer 97 menempati ruang-sesi nomer 35 yaitu hari selasa ruang lima sesi satu. Pada eksekusi II, terlihat angka 97 dan 22 artinya hari selasa ruang satu sesi dua. Angka 97, 98, dan 99 dalam contoh ini adalah satu orang dosen yang mengajar 3 mata kuliah. Maka hari yang diperoleh adalah hari selasa, rabu dan kamis sehingga waktu mengajar dosen tersebut tidak mungkin tabrakan.

KESIMPULAN

Berdasarkan hasil eksekusi dari pemrograman yang dilakukan, algoritma penjadwalan yang diusulkan berhasil melakukan penjadwalan untuk 100 dosen-matakuliah menggunakan 5 ruang, 4 sesi dan 5 hari kerja selama satu minggu. Semua ruang dan waktu yang disediakan dapat diisi dengan tepat tanpa terjadi tabrakan. Adapun untuk pengembangan selanjutnya, disarankan untuk melakukan penambahan-penambahan kriteria dan fungsi dari penelitian ini yaitu: perlu dikembangkan untuk prioritas mata kuliah yang sulit dan memerlukan konsentrasi yang tinggi dan energi yang besar sebaiknya dilaksanakan pada pagi hari. Penjadwalan akan dinilai optimal jika dapat menempatkan matakuliah yang di prioritaskan di pagi hari (atau session pertama)

DAFTAR PUSTAKA

- [1] Alade O. M., Omidiora E. O., dan Olabiyisi, S. O. (2014), *Development of a University Lecture Timetable using Modified Genetic Algorithms Approach*, International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 4, Hal. 163-168
- [2] Daskalaki, S., Birbas, T., dan Housos (2008) *School Timetabling for Quality Student and Teacher Schedules*, Journal Scheduling, Springer Science
- [3] Jha S.K. (2014) *Exam Timetabling Problem Using Genetic Algorithm*, International Journal Of Research In Technological Educational Institute of Serres, Greece Engineering and Technology, Vol. 03, Hal. 649-654
- [4] Kazarlis, S., Petridis V. dan Fragkou P. (2010) *Solving University Timetabling Problems Using Advanced Genetic Algorithms*, Research Report, [5]
- [6] Kanoh, H., dan Sakamoto, Y. (2008) *Knowledge Based Genetic Algorithm for University Course Timetabling Problems*, International Journal of Knowledge Based and Intelligent Engineering Systems, Hal. 283-294
- [7] Dipesh M, Hiral, D., Mohammed, S. dan Renuka, N. (2015) *Automatic Timetable Generation Using Genetic Algorithm*, International Journal of Advanced Research in Computer and Communication Engineering ,Vol. 4, Hal. 245-248
- [8] Haupt R.L. dan Haupt S.E. (2004) *Practical Genetic Algorithms*, John Wiley & Sons, New Jersey
- [9] Oluwasefunni, T. A., Adio T. A., dan Omotoyosi M. A (2010) *A Genetic Algorithm Approach for a Real World University Examination Timetabling Problem*, International Journal of Computer Application, Vol. 12, Hal. 1-4
- [10] Coulon, C. H. dan Harmelen, F. (1988) *Controlling Generate Test in Any Time*, Research Report, German National Research Institute for Computer Science