

BAB II

TINJAUAN PUSTAKA

A. ALGORITMA

Algoritma adalah metode langkah demi langkah pemecahan dari suatu masalah. Kata algoritma berasal dari matematikawan Arab ke sembilan, Al-Khowarizmi. Algoritma didasarkan pada prinsip-prinsip Matematika, yang memainkan peranan penting dalam Matematika dan Ilmu Komputer. Agar solusi masalah bisa dijalankan oleh komputer, solusi harus dijalankan sebagai serangkaian langkah-langkah yang tepat.

(Johnsonbough, 1998 : 135)

B. ALGORITMA GENETIKA

1. Sejarah Algoritma Genetika.

Algoritma genetika merupakan metode pencarian yang disesuaikan dengan proses genetika dari organisme biologi yang berdasarkan pada teori evolusi Charles Darwin. Landasan teoritis untuk algoritma genetika ini diajukan oleh John Holland (1975) yang kemudian dikembangkan lebih lanjut oleh muridnya David Gilberg. Algoritma genetik terinspirasi dari mekanisme seleksi alam, dimana individu yang lebih kuat kemungkinan akan menjadi pemenang dalam lingkungan yang kompetitif dan solusi yang lebih optimal dapat diperoleh dan diwakilkan oleh pemenang akhir dari permainan genetika. Dalam evolusi genetik kromosom yang lebih

sehat memiliki kecenderungan menghasilkan turunan yang lebih sehat juga. Dalam praktek penerapan algoritma genetika *kromosom* dari suatu populasi yang tersedia secara acak. Siklus operasional genetik akan berhasil apabila kromosom orang tua (*parent*) dipilih melalui rutinitas penyeleksian secara spesifik. Gen dari *parents* digabung untuk menghasilkan keturunan yang merupakan generasi baru dari proses evolusi ini diharapkan *kromosom* yang lebih baik akan menghasilkan jumlah keturunan yang lebih banyak dan mungkin berhasil bertahan pada generasi berikutnya.

(Nugroho, 2008)

2. Pengertian Algoritma Genetika.

Algoritma Genetika adalah algoritma dari proses pencarian solusi optimasi penggunaan pencarian acak, ini terlihat pada proses pembangkitan populasi awal yang menyatakan sekumpulan solusi yang terpilih secara acak. Algoritma ini memanfaatkan proses seleksi alamiah yang dikenal dengan proses evolusi. Dalam proses evolusi, individu secara terus menerus mengalami perubahan gen untuk menyesuaikan hidupnya. Hanya individu yang kuat yang mampu bertahan, sehingga dalam proses evolusi dapat diharapkan diperoleh individu yang terbaik. Proses alamiah ini melibatkan perubahan gen yang terjadi pada individu melalui proses perkembang biakan untuk memperoleh keturunan yang lebih baik.

Algoritma genetika merupakan program yang menstimulasikan proses evolusi. Dalam hal ini populasi dari *kromosom* dihasilkan secara random

dan memungkinkan untuk berkembang biak sesuai hukum-hukum evolusi dengan harapan menghasilkan individu *kromosom* yang prima. Kromosom ini pada kenyataan adalah kandidat penyelesaian masalah, sehingga bila *kromosom* yang baik berkembang, solusi yang baik terhadap masalah diharapkan akan dihasilkan.

Algoritma ini bekerja dengan sebuah populasi yang terdiri dari individu-individu, yang masing-masing individu mempresentasikan sebuah solusi yang mungkin bagi persoalan yang ada. Dalam kaitan ini individu dilambangkan dengan sebuah nilai *fitness* yang akan digunakan untuk mencari solusi yang terbaik dari persoalan yang ada.

Pertahanan yang tinggi dari individu memberikan kesempatan untuk melakukan reproduksi melalui perkawinan silang dengan individu yang lain dalam populasi tersebut. Individu baru yang dihasilkan dalam hal ini dinamakan keturunan, yang membawa beberapa sifat dari induknya. Sedangkan individu dalam populasi yang tidak terseleksi dalam reproduksi akan mati dengan sendirinya. Dengan jalan ini, beberapa generasi dengan karakteristik yang bagus akan bermunculan dalam populasi tersebut, untuk kemudian dicampur dan ditukar dengan karakter yang lain. Dengan mengawinkan semakin banyak individu, maka akan semakin banyak kemungkinan terbaik yang dapat diperoleh.

Sebelum Algoritma Genetika dapat dijalankan, maka sebuah kode yang sesuai (*representatif*) untuk persoalan harus di rancang. Untuk ini maka titik solusi dalam ruang permasalahan dikodekan dalam bentuk

kromosom atau string yang terdiri atas komponen genetik terkecil yaitu gen. Dengan teori evolusi dan teori genetika, di dalam penerapan Algoritma Genetika akan melibatkan beberapa operasi yaitu:

- a. Operasi Evolusi yang melibatkan proses seleksi (*selection*) di dalamnya.
- b. Operasi Genetika yang melibatkan operator pindah silang (*crossover*) dan mutasi (*mutation*).

(Nugroho, 2008)

3. Beberapa istilah penting yang perlu diperhatikan penyelesaian permasalahan dengan algoritma genetika.

- a. Genotype (*gen*), merupakan sebuah nilai yang menyatakan satuan dasar yang membentuk suatu arti tertentu dalam satu kesatuan gen yang dinamakan kromosom.
- b. Allele, merupakan nilai dari gen.
- c. Kromosom, merupakan gabungan gen-gen yang membentuk nilai tertentu.
- d. Individu, merupakan sekumpulan gen dalam sistem algoritma genetika. Individu bisa dikatakan sama dengan *kromosom*, yang merupakan kumpulan *gen* menyatakan satu nilai atau keadaan yang menyatakan salah satu solusi yang mungkin dari permasalahan yang diangkat.
- e. Populasi, merupakan sekumpulan individu yang akan diproses bersama dalam satu siklus populasi.

f. Generasi, menyatakan satu siklus proses evolusi atau satu iterasi di dalam algoritma genetika.

(Kusumadewi, 2005)

4. Kerangka Dasar Algoritma Genetika

Algoritma genetika merupakan bagian dari perhitungan *evolusioner*. Pada awal tahun 1970, John Holland merupakan orang pertama yang mengemukakan bahwa algoritma genetika merupakan program komputer yang meniru proses evolusi alam (Hui & Hsiao). Algoritma genetika dapat diaplikasikan pada berbagai jenis masalah karena membutuhkan sedikit pengetahuan tentang domain dan dikatakan juga bahwa telah tersedia suatu *heuristic* di dalamnya, yaitu *heuristic* yang didasarkan pada teori evolusi *The Survival of The Fittest* (Fang. H. L.: 1994) Spears dalam tesisnya juga menyatakan algoritma genetika merupakan suatu *heuristic* untuk masalah *NP-complete*. Mengacu Hui & Hsiao, kerangka dasar algoritma genetika yang sering disebut *Simple Genetic Algorithm* oleh John Holland dinyatakan sebagai berikut.

```
Simple Genetic Algorithm
Pembangunan populasi
Evaluasi populasi
While kriteria penghentian tidak dipenuhi
{
    Lakukan seleksi
    Lakukan persilangan dan mutasi
    Evaluasi populasi
}
```

a. Populasi Awal

Proses pencarian penyelesaian optimal, algoritma genetika menggunakan sekumpulan kandidat penyelesaian yang disebut dengan

populasi (Lee, H. C. : 2000). Untuk memulai operasinya algoritma genetika membutuhkan adanya populasi awal. Populasi awal tersebut dapat dibangun secara acak tetapi juga diperoleh dari seleksi sekunder (*secondary storage*) (Bambrik, L, : 1997). Pada skripsi ini populasi awal tersebut dibangun secara acak.

b. Representasi Kromosom

Algoritma genetika tidak beroperasi dengan kandidat penyelesaian asli dari suatu masalah, tetapi beroperasi dengan kandidat penyelesaian yang telah direpresentasikan dalam suatu kode yang disebut kromosom oleh (Hui & Hsiao). Pemilihan representasi kromosom terkait dengan tipe data dari program komputer (Bambrik, L, : 1997). Beberapa cara untuk mempresentasikan kandidat penyelesaian menjadi kromosom, antara lain *fixed-length string representation*, *vector of real numbers representation* dan *permutation representations* (Wright, A. H. : 2002).. Pada skripsi ini representasi kromosom yang digunakan adalah *fixed-length string representation* dimana setiap bagian string tersebut berisi suatu integer, (Mukherjee, S. N, : 2001) menyebut representasi tersebut sebagai *array of integer*.

c. Fungsi Penalti

Kemampuan untuk membedakan kualitas dari kromosom merupakan suatu hal penting. *Fitness* dapat menunjukkan seberapa baik kromosom. Penalti menunjukkan hal yang sebaliknya. (Mukherjee, S.

N, : 2001) menyatakan hubungan antara *fitness* dan *penalti* sebagai berikut :

$$fitness = \frac{1}{1 + penalti} \dots\dots\dots (1)$$

Hubungan tersebut sesuai dengan yang diungkapkan Ross et al, bahwa *fitness* berbanding terbalik dengan jumlah pelanggaran kendala pada suatu kromosom. Setiap pelanggaran kendala diklasifikasikan ke beberapa tipe dimana setiap tipe diberi bobot *penalti* berbeda sesuai seberapa penting kendala tersebut.

Misal C_j adalah himpunan kendala dengan tipe j . Setiap pelanggaran terhadap anggota C_j diberi bobot *penalti* w_j . Untuk setiap $c \in C_j$, didefinisikan

$$v(c) = \begin{cases} 1, & \text{jika terjadi pelanggaran kendala } c, \\ 0, & \text{jika tidak terjadi pelanggaran kendala } c, \end{cases}$$

Fungsi *fitnessnya* adalah

$$fitness = \frac{1}{(1 + \sum_{tipe\ j} w_j \sum_{c \in C_j} v(c))} \dots\dots\dots (2)$$

Dari persamaan (1) dan (2) diperoleh bahwa fungsi *penalti* adalah:

$$Penalti = \sum_{tipe\ j} w_j \sum_{c \in C_j} v(c) \dots\dots\dots (3)$$

5. Parameter Algoritma Genetika

Pengoperasian Algoritma Genetika dibutuhkan 3 parameter yaitu :

a. Probabilitas Persilangan (*Probability Crossover*)

Menunjukkan kemungkinan *crossover* terjadi antara 2 kromosom. Jika tidak terjadi *crossover* maka keturunannya akan sama persis dengan kromosom orang tua, tetapi tidak berarti generasi yang baru akan sama persis dengan generasi yang lama. Jika probabilitas *crossover* 100%, maka semua keturunannya dihasilkan dari *crossover*. *Crossover* dilakukan dengan harapan bahwa kromosom yang baru akan lebih baik.

(Obitko, M : 1998)

b. Probabilitas Mutasi (*Probability Mutation*)

Menunjukkan kemungkinan mutasi terjadi pada gen-gen yang menyusun sebuah kromosom. Jika tidak terjadi mutasi maka keturunan yang dihasilkan setelah *crossover* tidak berubah. Jika terjadi mutasi bagian kromosom akan berubah. Jika probabilitasnya 100% semua kromosom dimutasi. Jika probabilitasnya 0% tidak ada yang mengalami mutasi.

(Obitko, M : 1998)

c. Jumlah Populasi

Menentukan jumlah populasi atau banyaknya generasi yang dihasilkan, digunakan sebagai batas akhir proses seleksi, persilangan dan mutasi.

(Obitko, M : 1998)

6. Operator dalam Algoritma Genetika

Operator genetika dipergunakan untuk mengkombinasi (modifikasi) individu dalam aliran populasi guna mencetak individu pada generasi

berikutnya. Ada tiga operator genetika yaitu seleksi, persilangan (*crossover*) dan mutasi (*mutation*).

a. Seleksi

Individu di alam yang mempunyai karakter bertahan hidup lebih baik dapat bertahan untuk periode waktu yang panjang. Hal ini memberikan peluang yang lebih baik untuk menghasilkan keturunan. Dengan demikian setelah periode waktu yang panjang, seluruh populasi terdiri dari individu dengan karakter superior. Proses ini disebut seleksi alam (*natural selection*). (Dyer, C.)

Proses seleksi tersebut diterapkan dalam algoritma genetika dengan adanya operator seleksi. Operator seleksi berperan untuk membentuk *intermediat population*, yaitu populasi yang terdiri atas kromosom yang berpeluang untuk bereproduksi untuk menghasilkan populasi baru pada generasi mendatang (Whitley, D : 1993). Pada proses seleksi, kromosom yang lebih baik mempunyai peluang lebih besar untuk disalin ke dalam *intermediate population* yang disebut (Hui & Hsiao) sebagai *mating pool*. Ukuran dari *mating pool* sama dengan ukuran populasi. (Guthula) mengungkapkan cara kerja metode seleksi turnamen biner sebagai berikut :

- 1) Dua kromosom diambil secara acak dari seluruh anggota populasi untuk membentuk suatu turnamen .

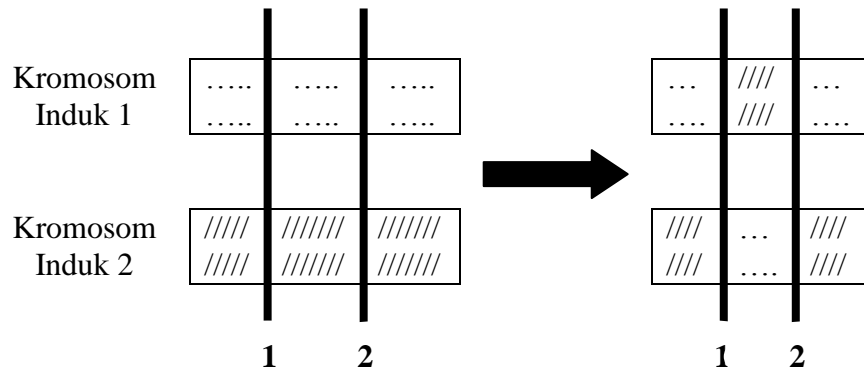
- 2) Kromosom tersebut dikompetensikan, dimana kromosom dengan *penalti* yang lebih rendah mempunyai kemungkinan yang lebih besar masuk *intermediate population*.
- 3) Kedua kromosom dikembalikan ke populasi dan turnamen yang lain dimulai.
- 4) Prosedur ini diulang terus sampai *intermediate population* penuh.

b. Persilangan (*crossover*)

Operator persilangan bertujuan mencampur komponen yang baik dari dua kromosom berbeda dengan harapan diperoleh kromosom yang terbaik. Persilangan diterapkan dengan probabilitas tertentu yang disebut probabilitas persilangan (Whitley, D : 1993). Sejalan dengan yang diungkapkan Whitley, Hui dan Haiiao, mengatakan bahwa tidak semua yang terpilih untuk reproduksi mengalami persilangan. Untuk itu diterapkan probabilitas persilangan yang dinotasikan dengan p_c . Jenis operator persilangan yang digunakan pada skripsi ini ada dua, yaitu persilangan dua titik (*two points crossover*) dan persilangan seragam (*uniform crossover*). Keduanya menjadi bagian dari persilangan banyak titik.

1) Persilangan Dua Titik

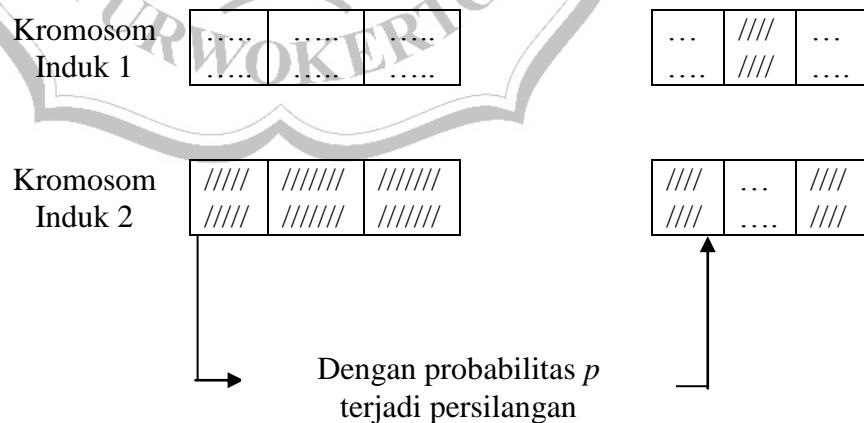
Pada persilangan ini dibutuhkan dua titik yang berguna untuk mengetahui posisi persilangan. Menurut (Fang, H. L.,:1994). persilangan dilakukan pada bagian kromosom yang terletak di antara dua titik tersebut seperti ditunjukkan oleh gambar 2.1



Gambar 2.1 Persilangan dua titik

2) Persilangan Seragam

Pada jenis operator ini persilangan dapat terjadi pada setiap bagian pada kromosom. Untuk mengendalikan persilangan tersebut digunakan suatu parameter p , dimana p adalah probabilitas bahwa *allele* kromosom induk pertama ditugaskan pada bagian kromosom yang bersesuaian pada kromosom induk kedua. Adapun proses persilangan ditunjukkan pada gambar 2.2



Gambar 2.2 Persilangan Seragam

7. Keuntungan Algoritma Genetika

Keuntungan Algoritma Genetika sangat jelas terlihat dari kemudahan implementasi dan kemampuan untuk menemukan solusi yang paling baik secara cepat untuk masalah-masalah berdimensi tinggi. Algoritma Genetika sangat berguna dan efisien untuk masalah dengan karakteristik sebagai berikut:

- a. Ruang masalah sangat besar, kompleks dan sulit dipahami.
- b. Kurang atau bahkan tidak ada pengetahuan yang memadai untuk mempresentasikan masalah ke dalam ruang pencarian yang lebih sempit.
- c. Tidak tersedianya analisis Matematika yang memadai.
- d. Ketika metode-metode konvensional sudah tidak mampu menyelesaikan masalah yang dihadapi.
- e. Solusi yang diharapkan tidak harus paling optimal, tetapi cukup bagus atau bisa diterima.
- f. Terdapat batasan waktu, misalnya dalam *real times system* atau sistem waktu nyata.

Algoritma Genetika telah banyak diaplikasikan untuk penyelesaian masalah dan pemodelan dalam bidang teknologi, bisnis dan entertainment seperti:

- a. Optimasi

Algoritma Genetika digunakan untuk optimasi numerik dan optimasi kombinatorial seperti *Traveling Salesman Problem* (TSP),

Perancangan *Integrated Circuit* atau IC (LOU93), *Lob Shop Schedulibg* (GOL91), optimasi video desain suara.

b. Pemrograman Otomatis

Algoritma Genetika telah digunakan untuk melakukan proses evolusi terhadap program komputer untuk merancang struktur komputasional, seperti *cellular automata* dan *sorting networks*

c. Machine Learning

Algoritma Genetika telah berhasil diaplikasikan untuk memprediksi struktur protein. Algoritma Genetika juga berhasil diaplikasikan dalam perancangan *neural networks* (jaringan syaraf tiruan) untuk melakukan proses evolusi terhadap aturan-aturan pada *learning classifier systems* atau *symbolic production systems*.

d. Model Ekonomi

Algoritma Genetika telah digunakan untuk memodelkan proses-proses inovasi dan pembangunan *building strategies*.

e. Model Sistem Imunisasi

Algoritma Genetika telah berhasil digunakan untuk memodelkan berbagai aspek pada system imunisasi alamiah, termasuk somatic mutation selama kehidupan individu dan menemukan keluarga dengan gen ganda (*multi-gen families*) sepanjang waktu evolusi.

f. Model Ekologis

Algoritma Genetika telah berhasil digunakan untuk memodelkan fenomena ekologis seperti *host-parasite co-evolution*, simbiosis dan aliran sumber daya dalam ekologi.

g. Interaksi antara Evolusi dan Belajar

Algoritma Genetika telah digunakan untuk mempelajari bagaimana proses belajar suatu individu bisa mempengaruhi proses evolusi suatu species dan sebaliknya.

(Suyanto, 2005 : 3 – 4

B. MASALAH PENJADWALAN

Penjadwalan secara umum adalah aktivitas memasukkan kegiatan pada jadwal. Jadwal adalah pembagian waktu berdasarkan rencana pengaturan urutan kerja atau daftar, tabel kegiatan, rencana kegiatan dengan pembagian waktu pelaksanaan yang rinci. Masalah penjadwalan sering dijumpai kendala, baik yang terhubung dengan tempat, waktu dan pelaksanaannya. Masalah ini muncul di berbagai bidang kegiatan maupun instansi seperti rumah sakit, universitas, sekolah, penerbangan, pabrik dan lain-lain. Desain penjadwalan itu bervariasi sesuai dengan kebutuhan serta keadaan kendala di lapangan.

Penjadwalan kegiatan belajar mengajar dalam suatu kampus adalah hal yang rumit. Terdapat berbagai aspek yang berkaitan dalam penjadwalan tersebut yang harus dilibatkan antara lain terdapat jadwal-jadwal di mana dosen yang bersangkutan tidak bisa mengajar. Tidak boleh adanya jadwal kuliah yang bersamaan dengan jadwal kuliah angkatan sebelumnya maupun

sesudahnya, sehingga mahasiswa dapat mengambil mata kuliah angkatan sebelumnya maupun sesudahnya. Distribusi jadwal perkuliahan juga diharapkan dapat merata tiap harinya untuk setiap kelas. Pekerjaan penjadwalan mata kuliah ini akan semakin berat jika melibatkan semakin banyak kelas per angkatan.

Masalah penjadwalan itu terlihat seperti masalah yang biasa yang dapat diselesaikan dengan metode pemikiran biasa, akan tetapi jika sudah dalam jumlah data yang banyak dan saling berkaitan maka akan memerlukan banyak waktu dan pikiran untuk menyelesaikannya. Adapun faktor yang menyebabkan adanya bentrokan dalam penyusunan sebuah jadwal adalah kondisi dosen, tempat kuliah yang terbatas, umlah mahasiswa cukup banyak, adanya berbagai kepentingan yang berbeda pada tiap orang dengan lokasi yang berbeda namun pada waktu yang sama serta kemampuan untuk menyusun jadwal. Selain itu faktor *human error* (kesalahan manusia) juga sangat menentukan proses penyusunan jadwal tersebut dapat terlaksana dengan baik.

(Aria, M. 2002)

C. INPUT DATA PENJADWALAN KULIAH

Terdapat empat masukan kelompok data yang perlu diberikan yaitu:

1. Tabel Mata Kuliah dengan Dosen.
2. Tabel Mata Kuliah dengan Semester.
3. Tabel Mata Kuliah dengan Bobot SKS.
4. Tabel Mata Kuliah dengan Kelas.

Tabel mata kuliah berisikan daftar seluruh mata kuliah yang akan dilaksanakan pada semester yang bersangkutan. Data yang perlu disertakan untuk setiap mata kuliah yang ada adalah kelas peserta, jumlah SKS, dosen pengajar serta ruangan mata kuliah tersebut.

Tabel Dosen, Tabel Kelas serta Tabel Ruang adalah tabel waktu yang menginformasikan waktu-waktu dari dosen, kelas maupun ruangan yang dapat digunakan untuk mata kuliah yang bersangkutan. Perlu dipahami bahwa yang dimaksud kelas adalah nama kelas mahasiswa misalnya kelas A, B dan C. Adapun yang dimaksud ruang adalah nama ruangan yang digunakan untuk perkuliahan misalnya menggunakan ruang G2.

(Aria, 2002 : 19 – 20)

D. MASALAH PENJADWALAN DI UMP

Jadwal mata kuliah merupakan hal yang sangat penting bagi kelancaran proses belajar mengajar khususnya di Program Studi Pendidikan Matematika UMP. Permasalahan yang sering terjadi yaitu banyaknya kelas di setiap angkatan, dosen yang terbatas dan jumlah mahasiswa yang banyak. Untuk menyelesaikan masalah penjadwalan khususnya di Program Studi Pendidikan Matematika UMP, maka harus ditetapkan dahulu *constraint* atau batasan dari masalah yang ada. *Constraint* dibagi menjadi dua yaitu *hard constraint* dan *soft constraint*. *Hard constraint* adalah batasan yang mutlak harus dipenuhi dalam proses penempatan jadwal. Sedangkan *soft constraint* adalah batasan dengan

prioritas lebih rendah yang masih memungkinkan untuk tidak diterapkan atau menjadi syarat mutlak dalam proses penempatan jadwal.

Adapun contoh dari *hard constraint* yang ada pada masalah penjadwalan kuliah khususnya di Program Studi Pendidikan Matematika UMP yaitu:

1. Tidak terdapat dosen yang mengajar mata kuliah yang berbeda pada waktu dan jam yang sama.
2. Waktu yang digunakan untuk perkuliahan setiap SKS selama 50 menit.
3. Perkuliahan setiap mata kuliah yang diambil 1 kali pertemuan dalam 1 minggu.
4. Ruang yang disediakan untuk perkuliahan khususnya Program Studi Matematika sebanyak 4 ruangan yaitu :
 - a. Ruang G.2 : maksimal 100 mahasiswa.
 - b. Ruang H.14 : maksimal 50 mahasiswa.
 - c. Ruang H.15 : maksimal 50 mahasiswa.
 - d. Ruang H.16 : maksimal 50 mahasiswa.
5. Hari aktif untuk perkuliahan adalah hari Senin hingga Sabtu.
6. Dalam 1 hari dibagi menjadi 5 jam pertemuan.

Sedangkan contoh dari *soft constraint* pada masalah penjadwalan khususnya di Program Studi Pendidikan Matematika UMP kuliah yaitu :

1. Dosen dapat memesan waktu mengajar tertentu yang diinginkan.
2. Penempatan jadwal untuk waktu yang telah dipesan dosen disesuaikan dengan prioritas dosen.