BAB II

LANDASAN TEORI

Untuk melakukan perancangan dan pembuatan aplikasi penjadwalan kuliah menggunakan Algoritma Tabu Search diperlukan pemahaman terhadap teori dan konsep yang mendasarinya, antara lain konsep mengenai penjadwalan, *algoritma Tabu Search,* khususnya dalam pembahasan Tabu Search sebagai kombinasi untuk masalah pengalokasian dan *Timetabling Subproblem* (TATI).

2.1 Penjadwalan

Jadwal adalah daftar output (tabel kegiatan) / rencana kegiatan yang harus dihasilkan dalam jangka waktu tertentu, biasanya disusun menurut urutan prioritas dengan pembagian waktu pelaksanaan yang terperinci sedangkan Penjadwalan adalah proses pembuatan / cara menjadwalkan suatu data menjadi jadwal (Harding,1984).

Menurut Marton (1993) Pendekatan terhadap metode penjadwalan secara garis besar dapat dibagi sebagai berikut

1. Penjadwalan dilakukan dengan manual

   Kelebihan dari penjadwalan yang dilakukan dengan manual adalah sebagai berikut:

   a. Cepat dan akurat dalam memadukan berbagai keahlian manusia dan prioritas.
   b. Mudah mengadaptasi perubahan.

   Kekurangan dari penjadwalan yang dilakukan manual adalah sebagai berikut :

SURABAYA
STIKOMP

a. Kurang mampu menguji sejumlah besar prioritas / keakuratan mengetahui dampak dari aktivitas yang berbeda
b. Kesulitan untuk menangani masalah yang komplek
c. Kesulitan menghadapi perubahan yang cepat (*future shock*)
d. Meningkatkan kekomplekan sistem (data *overload*)

2. Penjadwalan yang dilakukan dengan simulasi komputer

Kelebihan dari penjadwalan yang dilakukan dengan simulasi komputer adalah sebagai berikut:

a. Hampir sama dengan sistem nyata
b. Mudah dilakukan perubahan
c. Menunjukan pendekatan yang lebih alami dalam menyajikan keahlian manusia

Kekurangan dari penjadwalan yang dilakukan dengan simulasi komputer adalah sebagai berikut :

a. Hasil tidak selalu optimal
b. Metode pendekatan tidak selau tepat

3. Penjadwalan yang dilakukan dilakukan dengan metode matematis / deterministik, adalah sebagai berikut :

a) Integer dan Dynamic Programing

Kelebihan dari metode Integer dan Dynamic programing adalah hasil lebih baik sedangkan kekurangannya adalah perhitungan membutuhkan waktu yang lama

b) Neighbourhood Search / Non Deterministik (terdapat metode Tabu Search, Simulated Annealing, Genetyc Algorithm, dll)

STIKOMP SURABAYA

Kelebihan dari metode Neighbourhood Search adalah sebagai berikut :

a. Hasil yang di dapat relatif cepat
b. Keakuratan hasil tinggi

Kekurangan dari Metode Neighbourhood Search adalah sebagai berikut :

a. Keoptimalan hasil merupakan local optimal, bukan global optimal
b. Adanya sedikit intuisi dalam mencapai solusi.

2.2 Konsep Sistem Kredit Semester

Pelaksanaan pendidikan dan pengajaran menggunakan Sistem Kredit Semester yaitu penyelenggaraan program pendidikan tinggi untuk menyatakan beban studi mahasiswa, beban tugas edukatif, dan beban penyelenggaraan program.

Berdasarkan keputusan Menteri Pendidikan Nasional Indonesia Nomor 232/U/2000 tentang Pedoman Penyusunan Kurikulum Pendidikan Tinggi dan Penilaian Hasil Belajar Mahasiswa.

Pasal 1 nomor 12 : "Sistem Kredit Semester adalah sistem penyelenggaraan pendidikan dengan menggunakan Satuan Kredit Semester (SKS) untuk menyatakan beban studi mahasiswa, beban kerja dosen, pengalaman belajar dan beban penyelenggaraaan program".

Pasal 1 nomor 13 : "Semester adalah satuan waktu kegiatan yang terdiri atas 16 sampai 19 minggu kuliah atau kegiatan terjadwal lainnya, berikut kegiatan iringannya, termasuk 2 sampai 3 minggu kegiatan penilaian".

Pasal 1 nomor 14 : "Satuan Kredit Semester selanjutnya disingkat SKS adalah takaran penghargaan terhadap pengalaman belajar yang diperoleh

STIKOM SURABAYA

selama satu semester melalui kegiatan terjadwal perminggu sebanyak 1 jam perkuliahan atau 2 jam pratikum, atau 4 jam kerja lapangan, yang masing-masing diiringi oleh skitara 1-2 jam kegiatan terstruktur dan skitar 1-2 jam kegiatan mandiri".

Berdasarkan rumusan ini dapat kita jabarkan hal-hal sebagai berikut:

1. SKS bertitik tolak dari pendekatan sistem. Sistem adalah suatu kesatuan yang terdiri dari atas komponen-komponen yang saling berhubungan, saling mempengaruhi dan saling bergantung satu sama lain untuk mencapai tujuan yang telah ditentukan. Secara konseptual, SKS terdiri atasa komponen-komponen input, proses, dan output. Inputnya adalah mahasiswa yang memiliki *entry behaviour* tertentu (motivasi, pengalaman, bakat, minat, kemampuan dan sebagainya).

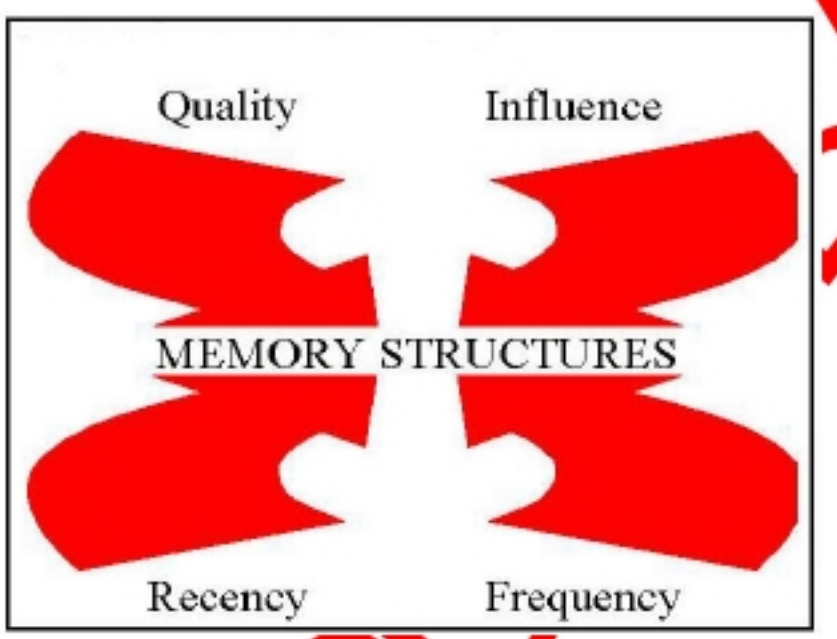
   Prosesnya adalah pengalaman belajar yang dipengaruhi oleh faktor-faktor Dosen (tenaga pengajar), program belajar, metode mengajar, sarana pendidikan dan waktu pendidikan. Outputnya adalah prestasi belajar dalam bentuk perangkat kemampuan yang diperoleh. Selain itu, juga tercakup komponen penilaian dan umpan balik.

2. SKS mempergunakan kredit sebagai ukuran atau patokan. Kredit mengandung makna "Penghargaan", yakni penghargaan terhadap (1)tercapainya peningkatan kemampuan (atau perfomance) yang diharapkan dengan asumsi bahwa mahasiswa yang bersangkutan telah memenuhi syarat-syarat atau program pendidikan yang telah ditetapkan; (2)pengalaman belajar dalam bentuk kegiatan-kegiatan kuliah dan

STIKOMP SURABAYA

(*counter*) yang akan selalu berubah setiap kali selesai dilakukan proses perulangan.

Struktur Tabu Search menggunakan 4 dimensi, terdiri dari *Recency*, *Frequency*, *Quality*, dan *Influence* yang ditujukan oleh gambar berikut :



Gambar 2.1 Dimensi Tabu Search

Recency-based dan *Frequency-based* memory saling melengkapi satu sama lain. Dimensi *Quality* berhubungan dengan kemampuan membedakan manfaat solusi yang diperoleh selama proses pencarian. Memory dapat mengidentifikasi elemen yang umum pada solusi yang baik atau pada path yang mengarah pada solusi yang baik. Dimensi *Influence* mempertimbangkan akibat pilihan yang dibuat selama proses pencarian tidak hanya terhadap kualitas maupun pada struktur.

Memori yang digunakan dalam Tabu Search adalah *eksplisit*, menyimpan solusi secara lengkap dan atributif, menyimpan informasi tentang atribut solusi yang berubah karena berpindah dari satu solusi ke solusi yang lain.

STIKOM PSS
JAYA

Komponen penting dari Tabu Search adalah strategi *Intensifikasi* dan *Diversifikasi*. Strategi *Intensifikasi* didasarkan pada modifikasi aturan pemilihan untuk mendorong kombinasi *move-move* dan ciri-ciri solusi terdahulu yang baik, menentukan kembali pada area yang menarik untuk diselidiki lebih dalam lagi. Memori eksplisit terkait erat dengan strategi *Intensifikasi*. Sebaliknya strategi *Diversifikasi* mendorong proses search untuk menyelidiki area yang belum dikunjungi.

Tabu Search dapat diaplikasikan langsung baik verbal atau pernyataan simbolik dari berbagai masalah keputusan tanpa harus mengubah dalam bentuk formulasi matematika untuk mengekspresikan secara lebih jelas ruang lingkup dan batasan masalah yang dihadapi. Permasalahan tersebut dapat dikarakteristik sebagai permasalahan optimasi (maksimum/minimum) dari Fungsi Tujuan f(x) dengan konstrain $x \in x$ dimana f(x) merupakan formulasi linier ataupun non linier dan satuan x merupakan himpunan variabel keputusan x pada konstrain yang ada.

Tabu Search dimulai dengan alur Neighbourhood Search yang diikuti secara iteratif dari satu solusi ke solusi lainnya hingga kriteria terminasi (Termination Criteria) yang ditetapkan sebelumnya dipenuhi tiap $x \in x$ memiliki hubungan dengan neighbourhood : $N(x) \subset x$.

Saat ini telah diterima secara luas bahwa perolehan kualitas solusi dan efisiensi algoritma dapat diporoleh dengan menggabungkan suatu variasi hal-hal tambahan penting agar pencarian didorong untuk berada pada area yang baik suatu proses yang dikenal sebagai *Intensifikasi*, tetapi juga diberi kesempatan yang cukup untuk memungkinkan *Diversifikasi* atau pergerakan menjauhi area saat ini, menuju area baru yang menjanjikan.

STIKOM SURABAYA

Banyak perkembangan baru memfokuskan pada mekanisme untuk melakukan hal ini secara lebih terarah daripada secara acak. Hal ini termasuk penggunaan suatu neighbourhood tertentu yang telah dipilih, dikenal sebagai strategi daftar calon (Candidat List); pengenalan hal-hal yang memainkan peran penting dalam membimbing proses pencarian menuju solusi yang baik, dikenal sebagai atribut atribut yang berpengaruh (Influental Atributes); dan pencarian yang diperkenankan bergerak maju mundur antara area yang berbeda didaerah solusi (Solution Space) dikenal sebagai strategi *Osilasi*.

Algoritma Tabu Search memiliki kemampuan untuk keluar dari *lokal optimal* tetapi Tabu Search tidak dapat menentukan global optimal. Tabu Search harus memiliki batasan maksimum Jumlah Iterasi, parameter *Maxit* dan ukuran Tabu List yang ditentukan sendiri oleh individu yang menggunakan metode ini / administrator sistem. Jumlah Iterasi yaitu banyaknya iterasi yang akan dilakukan untuk mengeksplorasi berbagai *Space Search Area*. Semakin besar jumlah maksimum iterasi, semakin besar pula peluang untuk menentukan solusi optimal secara global, namun memerlukan waktu perhitungan yang lama. Sedangkan parameter *Maxit* yang merupakan suatu batasan jumlah iterasi yang dilakukan pada suatu area tertentu mendorong suatu langkah *Diversifikasi* demi menghindari keterjebakan dalam suatu *Lokal Optimal*.

Secara garis besar elemen-elemen utama pada Tabu Search adalah sebagai berikut :

a. Representasi solusi : setiap *solusi feasible* pada suatu permasalahan optimasi harus direpresentasikan secara unik.

STIKOMP SURABAYA

b. Fungsi Cost : setiap fungsi cost (Fungsi tujuan) akan memetakan setiap solusi feasible ke nilai costnya.

c. Neighbourhood (tetangga) : suatu fungsi cost akan memetakan setiap *solusi fasible* S ke solusi-solusi yang lainnya.

d. Tabu List : suatu list yang berisi T gerakan terakhir.

e. Jumlah elemen yang harus ada pada Suatu Solusi.

Algoritma Tabu Search secara garis besar dapat dituliskan sebagai berikut :

```
0. Tetapkan :
     • X=matriks input berukuran nxm
     • MaxItr=maksimum iterasi
1. S=bangkitkan solusi secara random.
2. GlobalMin=FCost(S).
3. Best=S.
4. TabuList = [].
5. Kerjakan dari k=1 sampai MaxItr :
6.          BestSoFar = FCost(S).
7.          BestMove=S.
8.          Kerjakan dari i=1 sampai (n-1)
9.               kerjakan dari j=i sampai n:
10.                    L= Tukar(S[i],S[j]).
11.                    Cost=FCost(L).
12.                    Jika (L∉ TabuList) atau (Cost < GlobalMin), kerjakan:
13.                         Jika (Cost<BestSoFar), kerjakan
14.                          BestSoFar=Cost.
15.                          BestMove=L.
16. S=BestMove.
17. Tambahkan S ke TabuList.
18. Jika BestSoFar < GlobalMin, Kerjakan:
19.         GlobalMin=BestSoFar.
20.         Best=BestMove.
```

Gambar 2.2 Algoritma Tabu Search

Solusi akhir adalah Best, dengan cost sebesar GlobalMin

PABAYA

2.3.1 Algoritma Tabu Search kombinasi untuk masalah pengalokasian

Tabu Search merupakan sebuah pendekatan *meta heuristik* yang biasanya digunakan untuk menemukan solusi yang mendekati optimal dengan jalan melakukan kombinasi dengan satu pendekatan deterministik yang berganti dari lokal optimal ke global optimal untuk memecahkan masalah penjadwalan dengan skala besar dan mengatasi sub masalah yang diformulasikan sebagai masalah pengalokasian. Pendekatan global disini menggunakan prosedur yang memodifiksi jadwal (*Timetable*) yang dilakukan secara berurutan sampai tidak ada lagi perbaikan dari fungsi tujuan.

Pendekatan global ini merupakan suatu prosedur heuristik dimana pada tiap iterasi, satu atau lebih *courses* dialokasikan kembali pada periode waktu yang berbeda. Demikian pula pada tiap Iterasi dihasilkan kembali pada *course section* yang bebeda sehingga model ini dihasilkan sebuah kondisi Local Optimum (tetapi umumnya mendekati global).

Tapi untuk masalah optimasi kombinatorial skala besar algoritma Tabu Search yang paling efisien dibandingkan metode heuristik lainnya karena Tabu Search ini memiliki daerah pencarian yang luas dan mencegah adanya pengulangan move sebelumnya.

2.3.2 Tabu Search for Timetabling Subproblem (TATI)

Setiap mata kuliah memiliki jumlah satuan kredit semester (SKS) yang berbeda- beda sehingga memerlukan pengaturan mata kuliah tersebut, selain itu harus diperhitungkan pula jumlah pengajar, jumlah ruangan yang tersedia dan lain-lain. Semua masalah tersebut merupakan obek-objek dari subproblem Timetabling.

STIKOMP SURABAYA

Timetable diindikasikan sebagai matrik misal Q_x x t , Untuk tiap mata kuliah x diisi kesemua waktu tiap mata kuliah t, sehingga pengalokasian x ke t menghasilkan *feisible timetabling* dan tiap mata kuliah x harus diperbaharui pada tiap langkah dari TATI karena x tergantung pada Timetable sekarang (*Current Timetable*). Kemudian solusi tetangga N(s) terdiri dari semua Timetable yang diperoleh dari pengalokasian sebuah mata kuliah x ke sebuah waktu tiap mata kuliah t baru.

Pada tiap langkah pada algoritma TATI dilakukan penyelidikan (*search*) langsung dengan membangkitkan sebuah sampel acak dari solusi tetangga dari sebuah Timetable x, sehingga kita dapat menghubungkan pengalokasian kembali bagian pelajaran yang menimbulkan sedikitnya suatu konflik dalam Timetable sekarang. Dengan cara seperti itu maka *feasible Timetable* tetangga yang tidak dapat memperbaiki fungsi tujuan tidak diperhitungkan.

Beberapa masalah yang terjadi dalam tiap langkah TATI adalah terjadinya konflik, yaitu konflik antara mata kuliah, dosen dan ruangan. Beberapa kondisi yang menimbulkan permasalahan(konflik) sebagai berikut :

a. Dalam satu satuan waktu penyelenggaraan kuliah tidak ada kelas (ruangan) yang digunakan oleh lebih dari satu Mata kuliah secara bersamaan.
b. Dalam satu satuan waktu penyelenggaraan kuliah tidak ada dosen yang mengajar lebih dari satu mata kuliah secara bersamaan.

STIKOMP SURABAYA