

Optimasi Perencanaan Pengambilan Mata Kuliah Dengan Metode Goal Stack Planning

Ni Wayan Sumartini Saraswati

STMIK STIKOM INDONESIA

Denpasar, Bali, Indonesia

geck_nik@yahoo.co.id

Abstrak

Metode *Goal Stack Planning* (GSP) telah diterapkan dalam menyusun balok dari keadaan awal yang acak menjadi keadaan yang diinginkan dengan tetap memenuhi syarat-syarat yang telah ditentukan sebelumnya.

Dengan menganalogikan sebuah mata kuliah sebagai sebuah balok maka dalam penelitian ini metode GSP akan digunakan untuk mengoptimasi perencanaan pengambilan mata kuliah.

Hasil perencanaan tersebut nantinya akan digunakan sebagai sebuah rekomendasi penunjang keputusan bagi mahasiswa ketika dihadapkan pada proses registrasi. Tujuan kedua adalah pembelajaran untuk mengimplementasikan metode *Goal Stack Planning* pada penyelesaian permasalahan di dunia nyata.

Keadaan awal adalah kondisi mata kuliah yang telah dilalui mahasiswa dan dinyatakan lulus. Kondisi akhir adalah rekomendasi perencanaan mata kuliah per semester yang akan ditempuh selanjutnya sehingga tujuan untuk lulus dengan cepat terpenuhi.

Kata kunci :

Goal Stack Planning, Optimasi Perencanaan, Registrasi Perkuliahan

1. PENDAHULUAN

Proses registrasi pengambilan mata kuliah umumnya dilakukan oleh mahasiswa tiap semester. Pada perguruan tinggi yang menerapkan sistem pengambilan mata kuliah dengan penawaran akan memberikan keleluasaan pada mahasiswa untuk menentukan sendiri mata kuliah yang akan ditempuh di semester berikutnya. Proses akan dilakukan mahasiswa dengan mengumpulkan informasi mata kuliah yang telah lulus, kemudian mengacu pada peta kurikulum mereka akan merancang pengambilan mata kuliah berikutnya. Kekurangan dari proses tersebut adalah, mahasiswa jarang melakukan perencanaan jangka panjang misalnya hingga semester dimana tugas akhir harus ditempuh. Berdasarkan pengamatan mahasiswa hanya akan melihat satu semester ke depan, padahal jika mahasiswa salah dalam melakukan pilihan maka akan terganjal di semester-semester berikutnya karena adanya sistem mata kuliah bersyarat.

Dua metode perencanaan yang cukup populer dan sudah pernah diuji pada Dunia Balok adalah Goal-Stack-Planning (GSP) dan Constraint-Posting

(CP). GSP dan CP memiliki kelemahan dan keunggulan masing-masing. Dari segi kemudahan implementasi biaya komputasi, GSP lebih unggul dibanding CP.

Berdasarkan pengamatan tersebut maka ingin dibentuk sebuah aplikasi yang mampu memberikan rekomendasi perencanaan optimum untuk pengambilan mata kuliah hingga semester terakhir kepada mahasiswa menggunakan metode Goal Stack Planning.

2. KAJIAN PUSTAKA

Planning adalah suatu metode penyelesaian masalah dengan cara memecah masalah ke dalam sub-sub masalah yang lebih kecil, menyelesaikan sub-sub masalah satu demi satu, kemudian menggabungkan solusi-solusi dari sub-sub masalah tersebut menjadi sebuah solusi lengkap dengan tetap mengingat dan menangani interaksi yang ada antar sub masalah.

Yang menjadi komponen *planning* adalah sebagai berikut :

- Memilih aturan terbaik untuk mengaplikasikan langkah berikutnya berdasarkan pada informasi heuristic terbaik yang tersedia
- Mengaplikasikan aturan terpilih untuk mengkomputasi state permasalahan baru yang timbul dari komputasi tersebut.
- Mendeteksi apakah solusi telah didapatkan
- Mendeteksi jalan buntu sehingga hal ini dapat dihindari
- Mendeteksi bilamana solusi yang hampir benar ditemukan dan mengerjakan teknik spesial sehingga solusi tersebut menjadi benar secara keseluruhan

2.1 Goal-Stack-Planning (GSP)

Dalam menyelesaikan sebuah masalah, GSP menggunakan sebuah stack untuk menampung kondisi-kondisi (kondisi goal dan kondisi-kondisi yang mungkin terjadi ketika pencarian solusi) dan operator-operator yang telah diajukan untuk memenuhi kondisi-kondisi tersebut. Juga, tergantung pada sebuah basis data yang menggambarkan current-state dan satu set operator yang dideskripsikan sebagai daftar predikat PRECONDITION, ADD dan DELETE atau daftar-PAD.

Langkah – langkah dalam menyelesaikan masalah menggunakan GSP

1. Langkah pertama GSP dalam menyelesaikan sebuah masalah adalah menempatkan kondisi-kondisi goal-state pada stack. Kondisi-kondisi tersebut akan disimpan di dalam sebuah slot stack.
2. Langkah kedua, mengacu pada current -state, kondisi-kondisi goal-state yang belum tercapai dimasukkan ke dalam stack, masing-masing menempati sebuah slot. GSP tidak memiliki aturan khusus yang mengatur urutan pemasukan ke dalam stack dari kondisi-kondisi yang belum tercapai tersebut
3. Langkah ketiga, slot terisi yang berada paling atas pada stack akan diperiksa. Hal-hal yang akan dilakukan bergantung pada kondisi slot tersebut. Kondisi yang mungkin terjadi pada slot tersebut adalah sebagai berikut:
 - a. Kondisi 1
Jika slot berisi kondisi yang sudah memenuhi current -state, tetapi slot tidak terletak di dasar stack dan juga tidak terletak di atas slot yang berisi operator, maka isi slot akan di-pop dari stack dan pemeriksaan dilanjutkan pada slot berikutnya.
 - b. Kondisi 2
Jika slot berisi kondisi yang belum memenuhi current -state maka isi slot akan di-pop dari stack. Kemudian, sebuah operator yang sesuai untuk mencapai kondisi tersebut akan dimasukkan ke dalam stack. Setelah itu, serangkaian kondisi yang dibutuhkan agar operator itu bisa diaplikasikan akan dimasukkan ke dalam sebuah slot stack. Selanjutnya, setiap kondisi dari rangkaian kondisi yang dibutuhkan operator agar dapat diaplikasikan tersebut akan dimasukkan ke dalam sebuah slot secara terurut, dimana kondisi yang harus dicapai paling akhir dimasukkan pertama kali.
 - c. Kondisi 3
Jika slot berisi kondisi atau rangkaian kondisi dan slot tersebut berada di atas slot yang berisi operator, maka isi slot teratas dari stack tersebut akan di-pop. Kemudian, operator pada slot berikutnya akan di-pop dan dimasukkan ke dalam antrian operator dalam rencana penyelesaian dan current -state di-update dengan mengaplikasikan operator tersebut pada current -state berdasarkan daftar-PAD.
 - d. Kondisi 4
Jika slot yang diperiksa adalah slot terdasar maka akan diuji kesamaan antara current - state dan goal-state. Jika sama (berarti goal-state telah tercapai) maka isi slot akan di- pop dan pencarian rencana penyelesaian dihentikan. Jika berbeda (goal-state belum tercapai) maka langkah ke dua diulangi. Jika kondisi yang terjadi bukan kondisi 4, setelah rangkaian tindakan yang bersesuaian dilakukan, langkah ketiga diulangi.

GSP mungkin menemui jalan buntu yang disadari (meningkatkan biaya komputasi) atau pun yang tidak disadari (membuat rencana penyelesaian tidak efisien). Jalan buntu yang tidak disadari disebabkan karena GSP tidak memiliki aturan untuk mengurutkan pemasukan kondisi goal-state yang belum tercapai ke alam stack (langkah ke-2).

Dalam metode ini, penyelesaian masalah memanfaatkan sebuah stack yang berisi operator dan goal yang telah disusun untuk memenuhi tujuan akhir pemecahan permasalahan tersebut. Penyelesaian masalah juga berdasarkan pada database yang menggambarkan situasi terkini dan sebuah himpunan operator yang menggambarkan list PRECONDITION, ADD dan DELETE. Metode goal stack planning menyerang permasalahan dengan melibatkan goal terkombinasi dengan cara memecahkan goal satu persatu dengan urutan tertentu. Sebuah plan (perencanaan) yang dibangun dengan metode ini berisi daftar berurutan dari sequence operator untuk mencapai goal yang pertama, diikuti oleh sequence yang lengkap dari goal kedua dan seterusnya.

3. PERCOBAAN DAN ANALISIS

3.1 Ruang Lingkup Software.

Dalam penelitian ini software dibangun menggunakan bahasa pemrograman Delphi 5, DBMS MySQL dengan interface berupa ODBC. Software yang dibuat berjalan dalam komputer *stand alone* yang tidak terhubung dengan jaringan komputer.

Input adalah daftar mata kuliah yang telah ditempuh diwujudkan dalam sebuah interface pemilihan mata kuliah oleh user. Input yang kedua berupa pemilihan semester terakhir yang telah ditempuh, seperti ditunjukkan oleh gambar 1.

Output berupa daftar mata kuliah yang direkomendasikan untuk ditempuh mahasiswa dalam semester selanjutnya, seperti ditunjukkan oleh gambar 2.

Data yang digunakan adalah kurikulum STMIK STIKOM Indonesia 2014 untuk program studi Manajemen Teknik Informatika. Mata kuliah diadakan di semester yang bersesuaian dalam artian mata kuliah di semester ganjil hanya ditawarkan di semester ganjil, begitu pula mata kuliah di semester genap hanya ditawarkan di semester genap.

3.2 Struktur data STACK

Berikut adalah struktur data yang digunakan dalam percobaan ini seperti ditunjukkan oleh tabel 1.

Tabel 1. Struktur data STACK

```

type
  elmt = ^elm;
  els = ^elt;
  els1 = ^elt1;
  els2 = ^elt2;
  elbool = ^boo;
  precond = record
    p1 : boolean;
    p2 : TList;
  end;
  elm = record
    op : string;
    p : precond;
    a : string;
  end;

  elt = record
    s : string;
  end;
  boo = record
    l : boolean;
    s : string;
    sms : integer;
    n : integer;
    op : string;
  end;

  elt1 = record
    s : string;
    n : integer;
    sms : integer;
  end;

  elt2 = record
    s : string;
    sms : integer;
  end;
    
```

Deklarasi variabel sebagai gambaran dapat ditunjukkan oleh table 2.

Tabel 2. Deklarasi Variabel

```

TEMP: els;
tamp : els1;
pad : TList;
tumpukan : TStack;
currentmk : TList;
currentsks : array [1..8] of integer;
IDX : INTEGER;
sksfix,sks : integer;
S : STRING;
pat : elmt;
rencana : TList;
bool : elbool;
    
```

Penjelasan dari beberapa variabel utama yang digunakan dalam program ini adalah sebagai berikut :

- a. Currentmk digunakan untuk menampung daftar mata kuliah yang telah lulus
- b. Tumpukan adalah stack yang berisi daftar semua mata kuliah yang akan diolah.
- c. Currentsks untuk menampung jumlah sks tiap semester yang telah lulus.

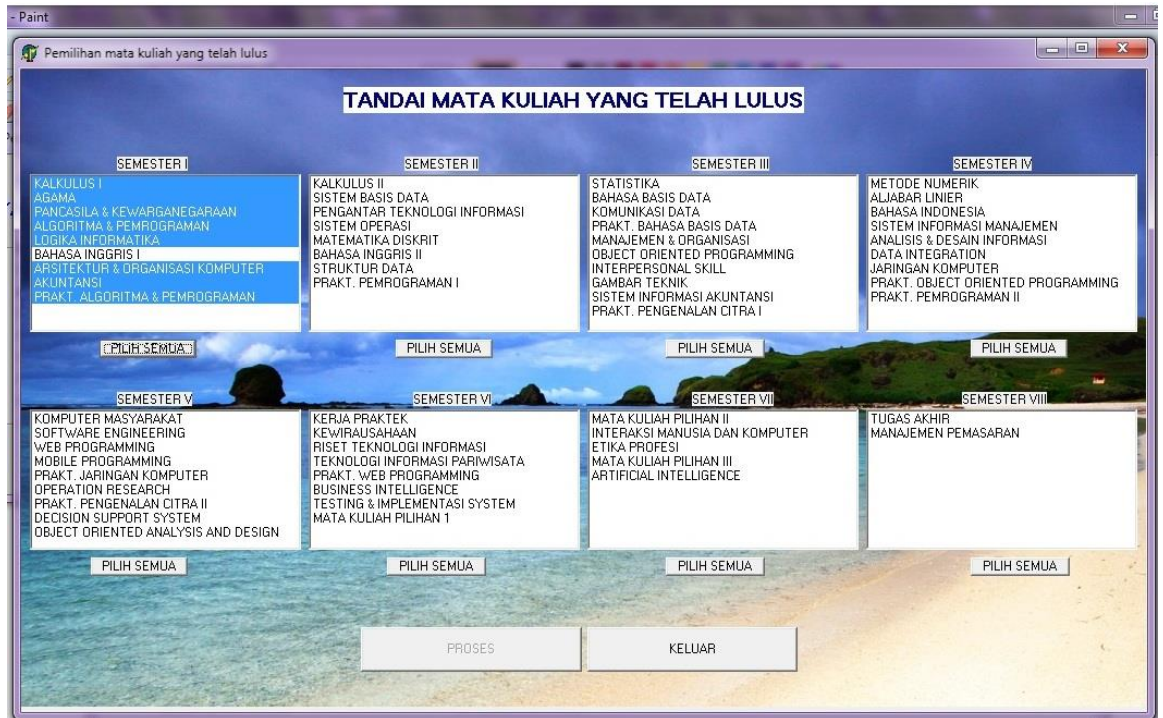
3.3 Percobaan

Sebuah percobaan dilakukan dengan input seperti ditunjukkan oleh gambar 1 dan posisi registrasi mahasiswa ada di semester 2 seperti ditunjukkan oleh gambar 2. Dalam percobaan ini mahasiswa telah lulus semua mata kuliah yang ditawarkan di semester I kecuali untuk mata kuliah Bahasa Inggris I.

Dibangkitkan juga mata kuliah bersyarat dimana mata kuliah Bahasa Inggris II tidak dapat ditempuh sebelum mata kuliah Bahasa Inggris I lulus. Hasil percobaan ditunjukkan oleh gambar 3. Dari hasil percobaan kita mendapatkan perencanaan studi dimana mahasiswa bisa menyelesaikan studi pada semester ke-7. Dalam daftar rencana hasil pengolahan menggunakan metode Goal Stack Planning mata kuliah Bahasa Inggris I direncanakan diambil di semester ke-3 dan mata kuliah Bahasa Inggris II direncanakan akan ditempuh di semester ke-4.

Berikut ini adalah cuplikan tampilan software perencanaan pengambilan mata kuliah menggunakan metode Goal Stack Planning ditunjukkan oleh gambar 1, gambar 2 dan gambar 3.

Sumartini, Optimasi Perencanaan Pengambilan Mata Kuliah Dengan Metode Goal Stack Planning



Gambar 1. Input Mata Kuliah



Gambar 2. Input Semester



Gambar 3. Output Software

4. SIMPULAN

Penelitian ini menunjukkan bahwa metode Goal Stack Planning dapat digunakan untuk memecahkan permasalahan perencanaan pengambilan mata kuliah mahasiswa STMIK STIKOM Indonesia.

REFERENCES

- [1] STMIK STIKOM Indonesia , Buku Pedoman Akademik 2013-2014 Program Studi Teknik Informatika dan Sistem Komputer, 2013.
- [2] STMIK STIKOM Indonesia ,Pohon Mata Kuliah TI, 2013.
- [3] Suyanto, Artificial Intelligence : Searching – Reasoning – Planning – Learning. Penerbit Informatika. 2014.
- [4] Jogiyanto, Analisis & Desain Sistem Informasi: Pendekatan Terstruktur Teori dan Praktik Aplikasi Bisnis. Andi Offset. 2005.
- [5] Ian Sommerville, Software Engineering, China Machine Press. 2006.
- [6] Kendall & Kendall, *Systems Analysis and Design*. Prentice Hall. 2011.