

# Penerapan Graf pada Constraints Satisfaction Problem

Ahmad Rizdaputra and 13513027  
Program Studi Teknik Informatika  
Sekolah Teknik Elektro dan Informatika  
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia  
13513027@std.stei.itb.ac.id

**Abstrak**—Matematika diskrit adalah salah satu mata kuliah jurusan teknik informatika yang bisa dikatakan adalah dasar dari hampir semua mata kuliah yang dipelajari di teknik informatika. Matematika diskrit mempelajari semua aspek matematik di dalam dunia ini secara diskrit yaitu tidak kontinu. Salah satu penerapannya di ilmu informatika adalah di dalam mata kuliah intelegensia buatan. Dalam intelegnsia buatan sangat banyak penerapan dari mata kuliah matematika diskrit salah satunya adalah graf. Graf secara singkat dapat digunakan untuk merepresentasikan objek-objek diskrit dan hubungan antara objek-objek tersebut. Intelegensia buatan, atau yang sering disebut juga Artificial Intelligence, adalah mata kuliah yang mempelajari tentang membuat sebuah program yang dapat menyerupai manusia baik untuk menyelesaikan masalah maupun untuk berperilaku seperti manusia. Graf di dalam intelegensia buatan diterapkan salah satunya pada CSP atau Constraints Satisfaction Problem.

**Kata Kunci**—CSP, graf.

## I. PENDAHULUAN

Matematika diskrit adalah salah satu cabang ilmu matematika yang mempelajari objek-objek diskrit, yaitu objek yang terdiri dari banyak elemen dan elemen-elemen tersebut tidak saling berhubungan.

Ilmu matematika diskrit dapat diterapkan ke berbagai ilmu teknik, ekonomi, sains murni, dan kehidupan sehari-hari secara umum. Penerapannya juga bisa di dalam ilmu informatika sendiri, yaitu mata kuliah lainnya yang dipelajari di teknik informatika. Salah satu penerapannya di dalam inmu informatika adalah pada intelegensia buatan atau *Artificial Intelligence*.

*Artificial Intelligence* adalah salah satu mata kuliah yang dipelajari di teknik informatika. Mata kuliah ini mempelajari cara membuat program yang dapat berperilaku seperti manusia baik dalam segi pemecahan masalah maupun dalam perilaku sehari-hari. *Artificial Intelligence* sedang berkembang dengan sangat pesat dan pemakaiannya sudah sangat meluas dan hamper ada di seluruh penjuru dunia.

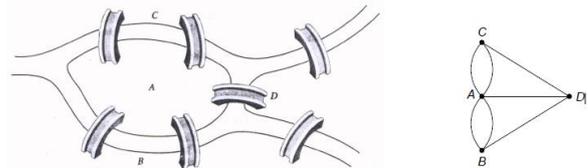
Penerapan matematika diskrit di dalam mata kuliah intelegensia buatan adalah pada bagian pemecahan masalah, dalam hal ini CSP atau *Constraints Satisfaction Problem*. Dalam CSP dibutuhkan penerapan dari materi

dasar tentang graf yang telah dipelajari di mata kuliah matematika diskrit.

## II. GRAF

Penemu graf adalah L. Euler ( Leonhard Euler ). Graf ditemukan disebuah jembatan Königsberg (tahun 1736). Di kota Königsberg (sebelah timur negara bagian Prussia, Jerman), yang sekarang bernama kota Kaliningrad, terdapat sungai Pregal yg mengalir mengitari pulau Kneiphof lalu bercabang menjadi dua buah anak sungai. Ada 7 buah jembatan yg menghubungkan daratan yg dibelah oleh sungai tersebut.

Berikut adalah gambar dari jemabatan Königsberg:



Gambar 2.1

Sumber:

<https://imeldaflorensia91.wordpress.com/2013/05/04/graf-matematika-diskrit/>

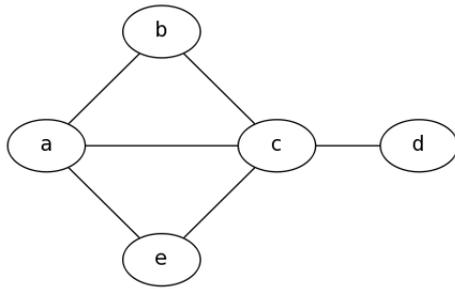
Diakses pada 12 Desember 2015 pukul 20.00

Suatu graf adalah himpunan benda-benda yang disebut simpul (*vertex* atau *node*) yang terhubung oleh sisi (*edge*) atau busur (*arc*). Biasanya graf digambarkan sebagai kumpulan titik-titik (melambangkan simpul) yang dihubungkan oleh garis-garis (melambangkan sisi) atau garis berpanah (melambangkan busur). Suatu sisi dapat menghubungkan suatu simpul dengan simpul yang sama. Sisi yang demikian dinamakan gelang (*loop*).

Adapun beberapa jenis graf yang dapat dibagi berdasarkan beberapa hal yaitu:

- Berdasarkan ada tidaknya gelang atau sisi ganda pada suatu graf, maka graf digolongkan menjadi dua jenis:
  1. Graf sederhana (*simple graph*).

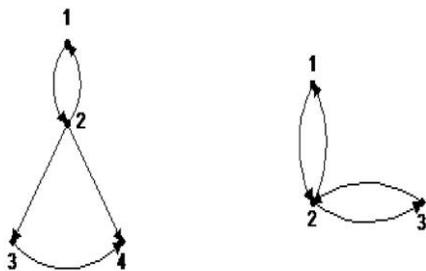
Graf yang tidak mengandung gelang maupun sisi-ganda dinamakan graf sederhana. Berikut adalah contoh dari gambar graf sederhana:



Gambar 2.2

Sumber: <http://graphs.grevian.org/example>  
Diakses pada 12 Desember 2015 pukul 20.30 WIB

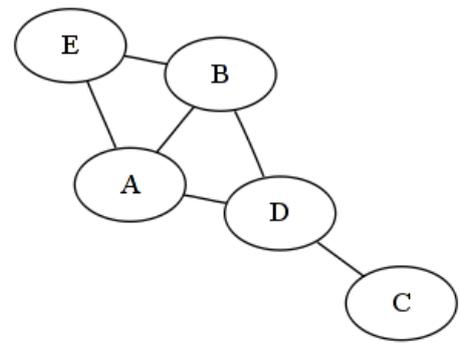
2. Graf tak-sederhana (*unsimple-graph*).  
Graf yang mengandung sisi ganda atau gelang dinamakan graf tak-sederhana (*unsimple graph*). Berikut adalah contoh dari gambar graf tak sederhana:



Gambar 2.3

Sumber: fheet.blogspot.com  
Diakses pada 12 Desember 2015 pukul 20.30 WIB

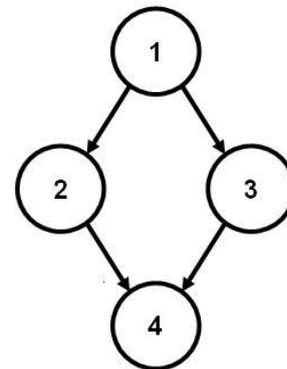
- Berdasarkan jumlah simpul pada suatu graf, maka secara umum graf dapat digolongkan menjadi dua jenis:
  1. Graf berhingga (*limited graph*)  
Graf berhingga adalah graf yang jumlah simpulnya,  $n$ , berhingga.
  2. Graf tak-berhingga (*unlimited graph*)  
Graf yang jumlah simpulnya,  $n$ , tidak berhingga banyaknya disebut graf tak-berhingga.
- Berdasarkan orientasi arah pada sisi, maka secara umum graf dibedakan atas 2 jenis:
  1. Graf tak-berarah (*undirected graph*)  
Graf yang sisinya tidak mempunyai orientasi arah disebut graf tak-berarah.



Gambar 2.4

Sumber: cs.stackexchange.com  
Diakses pada 12 Desember 2015 pukul 20.30 WIB

2. Graf berarah (*directed graph* atau *digraph*)  
Graf yang setiap sisinya diberikan orientasi arah disebut sebagai graf berarah.

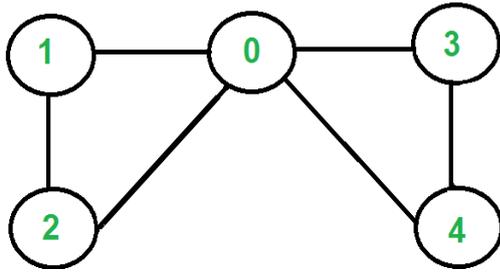


Gambar 2.5

Sumber: commons.wikimedia.org  
Diakses pada 12 Desember 2015 pukul 20.30 WIB

Adapun beberapa macam graf:

1. Graf Euler  
Lintasan Euler ialah lintasan yang melalui masing-masing sisi di dalam graf tepat satu kali. Sirkuit Euler ialah sirkuit yang melewati masing-masing sisi tepat satu kali. Graf yang mempunyai sirkuit Euler disebut graf Euler (*Eulerian graph*).  
Graf yang mempunyai lintasan Euler dinamakan juga graf semi-Euler (*semi-Eulerian graph*)



The graph has Eulerian Cycles, for example "2 1 0 3 4 0 2"  
Note that all vertices have even degree

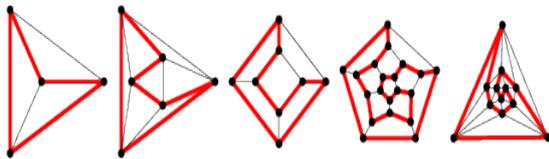
Gambar 2.6

Sumber: [www.gfgreader.info](http://www.gfgreader.info)

Diakses pada 12 Desember 2015 pukul 20.30 WIB

## 2. Graf Hamilton

Lintasan Hamilton ialah lintasan yang melalui tiap simpul di dalam graf tepat satu kali. Sirkuit Hamilton ialah sirkuit yang melalui tiap simpul di dalam graf tepat satu kali, kecuali simpul asal (sekali pun simpul akhir) yang dilalui dua kali. Graf yang memiliki sirkuit Hamilton dinamakan graf Hamilton, sedangkan graf yang hanya memiliki lintasan Hamilton disebut graf semi-Hamilton.



Gambar 2.7

Sumber: [mathworld.wolfram.com](http://mathworld.wolfram.com)

Diakses pada 12 Desember 2015 pukul 21.00 WIB

## C. Graf Isomorfik

- Dua buah graf yang sama tetapi secara geometri berbeda disebut graf yang saling isomorfik.
- Dua buah graf,  $G_1$  dan  $G_2$  dikatakan isomorfik jika terdapat korespondensi satu-satu antara simpul-simpul keduanya dan antara sisi-sisi keduanya sedemikian sehingga hubungan kebersisian tetap terjaga.
- Dengan kata lain, misalkan sisi  $e$  bersisian dengan simpul  $u$  dan  $v$  di  $G_1$ , maka sisi  $e'$  yang berkoresponden di  $G_2$  harus bersisian dengan simpul  $u'$  dan  $v'$  yang di  $G_2$ .
- Dua buah graf yang isomorfik adalah graf yang sama, kecuali penamaan simpul dan sisinya saja yang berbeda. Ini benar karena sebuah graf dapat digambarkan dalam banyak cara.

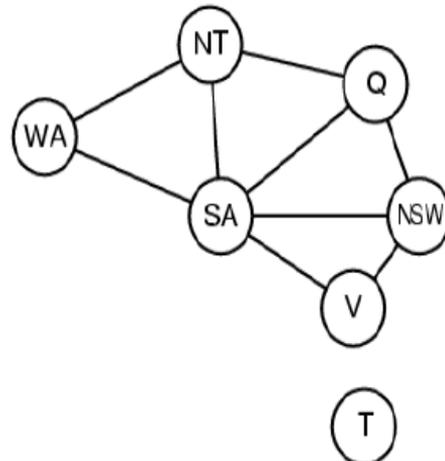
## III. CSP

CSP atau *Constraints Satisfaction Problem* adalah permasalahan yang tujuannya untuk mendapatkan suatu kombinasi variabel-variabel tertentu yang memenuhi aturan-aturan (*constraints*) tertentu.

Di dalam CSP ada beberapa komponen yaitu:

- *State*: didefinisikan dengan *variables*  $X_i$  yang mempunyai *values* dari domain  $D_i$ .
- *Goal Test*: adalah sebuah himpunan *constraints* yang memberikan kombinasi yang diizinkan untuk mengisi variabel.

Di dalam CSP diterapkan satu graf yang bernama *constraint graph* yaitu representasi di mana *node* adalah *variable*, *edge* adalah *constraint*. Salah satu contoh permasalahan yang dapat diselesaikan dengan *constraint graph* yaitu *coloring graph* untuk peta agar Negara yang bertetangga memiliki warna yang berbeda.



- Variables WA, NT, Q, NSW, V, SA, T
- Domains  $D_i = \{\text{red, green, blue}\}$
- Constraints: adjacent regions must have different colors
- e.g.,  $WA \neq NT$ , or  $(WA, NT)$  in  $\{(\text{red, green}), (\text{red, blue}), (\text{green, red}), (\text{green, blue}), (\text{blue, red}), (\text{blue, green})\}$

Adapun beberapa algoritma penyelesaian yang dapat digunakan untuk masalah ini:

### 1. Algoritma *Backtracking*

- Algoritma backtracking search (penelusuran kembali) adalah suatu bentuk algoritma *depth-first-search*.
- Jika solusi partial melanggar *constraint*, backtracking melakukan langkah kembali ke solusi partial sebelumnya.

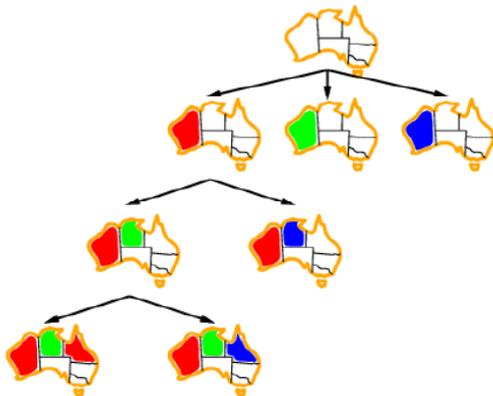
- Pada algoritma backtracking, teknik look ahead digunakan untuk meramalkan efek pemilihan variabel branching untuk mengevaluasi nilai-nilai variabel tersebut.
- Forward checking adalah salah satu cara untuk melakukan look ahead. Forward checking mencegah terjadinya konflik dengan melarang nilai-nilai yang belum diisikan ke variable untuk dipakai.
- Ketika suatu nilai diisikan ke suatu variabel, nilai yang berada di domain dari variabel yang konflik tersebut akan dihilangkan dari domain.

```

function BACKTRACKING-SEARCH(csp) returns a solution, or failure
  return RECURSIVE-BACKTRACKING({}, csp)

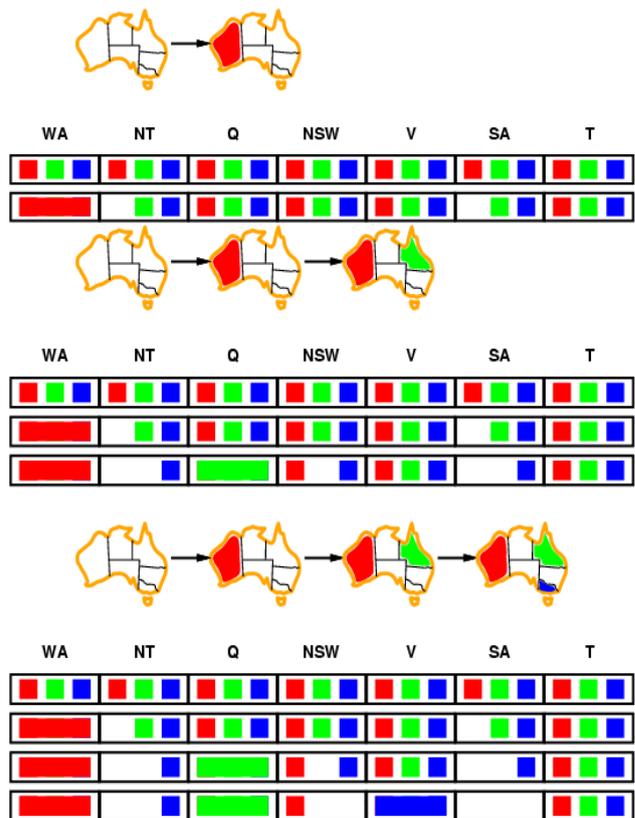
function RECURSIVE-BACKTRACKING(assignment, csp) returns a solution, or failure
  if assignment is complete then return assignment
  var ← SELECT-UNASSIGNED-VARIABLE(Variables[csp], assignment, csp)
  for each value in ORDER-DOMAIN-VALUES(var, assignment, csp) do
    if value is consistent with assignment according to Constraints[csp] then
      add { var = value } to assignment
      result ← RECURSIVE-BACKTRACKING(assignment, csp)
      if result ≠ failure then return result
      remove { var = value } from assignment
  return failure

```



## 2. Forward Checking

- Catat kemungkinan nilai sah untuk semua variable yang belum di-assign. Jika ada sebuah variable yang tidak ada kemungkinan nilai sah, langsung failure (backtrack).
- Simpan nilai valid untuk variable yang belum di-assign.
- Bila salah satu variable tidak mempunyai kemungkinan nilai yang valid maka search dihentikan.



## IV. SIMPULAN

Penerapan matematika diskrit sangatlah luas. Dan mata kuliah ini adalah landasan dari kuliah-kuliah seperti strategi algoritma dan intelegensia buatan. Penerapannya pada intelegensia buatan dapat dilihat salah satu nya dalam pemanfaatan graf dengan tau arti ketetanggaan dan aturan-aturan graf yang normal. Selain itu bukan hanya graf namun struktur pohon dan kompleksitas algoritma banyak diteraptakan di mata kuliah strategi algoritma dan intelegensia buatan

## REFERENCES

- [1] <http://danyatriokintoko.blogspot.co.id/>  
Waktu akses: 12 Desember 2015, 20.00 WIB
- [2] <http://dantikpuspita.com/konsep-dan-pengertian-dasar-graph-graf/>  
Waktu akses: 12 Desember 2015, 20.00 WIB
- [3] <http://oestadnetral.blogspot.co.id/2012/11/graf.html>  
Waktu akses: 12 Desember 2015, 20.00 WIB
- [4] <http://wrepaiwank.blogspot.co.id/2011/01/jenis-jenis-graf.html>  
Waktu akses: 12 Desember 2015, 20.00 WIB
- [5] [http://sejarahgraph.blogspot.co.id/2011/04/sejarah-graph\\_11.html](http://sejarahgraph.blogspot.co.id/2011/04/sejarah-graph_11.html)  
Waktu akses: 12 Desember 2015, 20.00 WIB
- [6] <https://imeldaflorensia91.wordpress.com/2013/05/04/graf-matematika-diskrit/>  
Waktu akses: 12 Desember 2015, 20.00 WIB
- [7] <http://graphs.grevian.org/example>  
Waktu akses: 12 Desember 2015, 20.30 WIB
- [8] <http://mathworld.wolfram.com>  
Waktu akses: 12 Desember 2015, 21.00 WIB
- [9] Slide Kuliah AI  
Waktu akses: 12 Desember 2015, 23.00 WIB
- [10] Slide Kuliah Matematika Diskrit  
Waktu akses: 12 Desember 2015, 23.00 WIB

## PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 10 Desember 2015

Ahmad Rizdaputra 13513027