PENENTUAN *MATCHING* MAKSIMUM PADA GRAF BIPARTIT BERBOBOT MENGGUNAKAN METODE HUNGARIAN

Muchammad Abrori[1] dan Rina Wahyuningsih[2]

Abstrak:
Matching is a part of graph theory that discuss to make a pair, that can be used to solve many problems; one of them is the assignment problem. The assignment problem is to make a pair problem for n as the employees and for n as the duties, therefore each employee gets one duty, and each duty is given exactly for each employee. The assignment problem can be solved by determining the matching in weighted bipartite graph through Hungarian Method. It can be determined from the alternating tree of a formed edge. If there is augmenting path, that augmenting path is used to form the more number of matching. If the formed path is alternating path, therefore the process is labeling the new node until finding the augmenting vertices. This matching is called as the perfect matching with the number of maximum weighed side in weighted bipartite graphs. The result matching is the solution for the assignment problem by giving an employee with a duty.

Keywords: matching, graph, assignment problem, Hungarian method.

PENDAHULUAN

Sebuah matching merupakan masalah mengenai mengawankan elemen-elemen dalam sebuah himpunan dengan elemen-elemen dalam himpunan yang lain. Sebuah matching merupakan subset M dari himpunan edge E sedemikian hingga tidak ada dua edge yang incident ke suatu verteks yang sama (Magun: 2000). Matching dapat diapliksikan dalam banyak hal, misalnya identifikasi keaslian tanda tangan, penentuan jadwal, solusi marriage problem atau pemasangan antara laki-laki dengan perempuan, dan penyelesaian dari masalah penugasan (assignment problem). Penempatan awal task-task ke dalam prosessor-prosessor, dimana biaya dari suatu penugasan merupakan fungsi total waktu eksekusi dan total waktu komunikasi sehingga diperoleh penugasan yang optimal yaitu penugasan yang meminimumkan fungsi kedua biaya tersebut (Venkateswaran: 1993) merupakan contoh lain aplikasi dari matching.

Masalah penugasan merupakan permasalahan memasangkan n pegawai dan n tugas, sedemikian sehingga setiap pegawai mendapatkan satu tugas, dan tiap tugas diberikan tepat pada satu pegawai. Masalah penugasan dapat direpresentasikan dalam bentuk graf bipartit. Graf bipartit merupakan graf yang tidak memiliki cycle ganjil, loop, dan dapat dipartisi menjadi dua bagian himpunan simpul yaitu V_1 dan V_2, dengan V_1 menunjukkan himpunan pegawai, sedangkan himpunan tugas ditunjukkan

[1] Jurusan Matematika, Universitas Islam Negeri Sunan Kalijaga
Jln. Adi Sucipto no. 1, Jakarta 11440
E-mail : borymuch@yahoo.com

[2] Jurusan Matematika, Universitas Islam Negeri Sunan Kalijaga
Jln. Adi Sucipto no. 1, Jakarta 11440

Naskah diterima: 21 Desember 2011, direvisi: 20 Maret 2012, disetujui: 15 Mei 2012

dengan V_2. Sebuah sisi graf menghubungkan sebuah pegawai dengan tugas dapat diberi sebuah bobot yang menunjukkan bobot atau nilai pegawai terhadap tugas.

Masalah penugasan dapat dicari hasil optimalnya dengan cara menentukan matching maksimum pada graf bipartit berbobot. Matching maksimum pada suatu graf bipartit berbobot adalah matching yang memiliki jumlah bobot yang maksimum, sehingga penentuan matching maksimum pada graf bipartit berbobot bukan hanya sekedar mencari matching dengan kardinalitas terbanyak, tetapi juga memperhatikan bobot yang termuat pada sisi graf bipartit tersebut. Cara menentukan matching maksimum pada graf bipartit berbobot ini dapat diselesaikan menggunakan metode Hungarian. Sehingga tujuan dari penelitian ini adalah untuk mendapatkan matching maksimum pada graf bipartit berbobot menggunakan metode Hungarian

Metode Hungarian merupakan salah satu metode yang digunakan untuk menyelesaikan masalah penugasan. Ada dua macam intrepretasi dari metode ini, yaitu dengan menggunakan matrik dan graf bipartit berbobot.

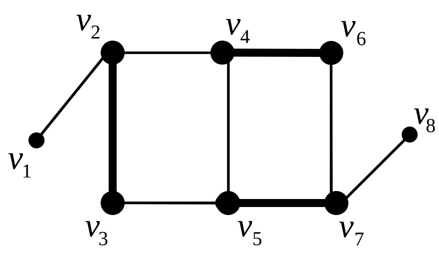
METODE PENELITIAN

Penelitian dimulai dengan mempelajari konsep dasar yang berkaitan dengan matching pada graf bipartit berbobot, metode Hungarian, dan masalah penugasan. Selanjutnya merumuskan masalah penugasan dalam bentuk graf bipartit berbobot, sehingga dapat diselesaikan menggunakan metode Hungarian. Sebuah matrik ukuran 5x5 disajikan sebagai contoh kasus penelitian (Howard Anton, 1987: 84). Matrik tersebut dibawa menjadi matrik penugasan dan direpresentasikan ke dalam bentuk graf bipartit berbobot, kemudian dicari matching berbobot maksimumnya menggunakan metode Hungarian.

MATCHING

Definisi : *Matching* (Kocay dan Kreher, 2004: 139)

Matching M adalah himpunan sisi sedemikian sehingga tidak terdapat dua sisi pada M yang bertemu pada simpul yang sama. Banyaknya *matching* yang terdapat pada suatu graf disebut kardinalitas *matching,* dan dilambangkan dengan $|M|$. Contoh: Sisi *matching* M_4 pada graf G_4 adalah sisi (v_2, v_3), (v_4, v_6) dan (v_5, v_7).



Terdapat dua istilah simpul pada matching:

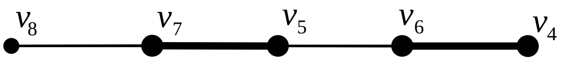
1. ***Simpul Saturated***. Simpul *saturated* adalah simpul yang incident dengan sisi pada matching M. (Gondran dan Minoux, 1984: 280)

   Contoh : simpul v_2, v_3, v_4, v_6
2. ***Simpul Unsaturated***. Simpul *unsaturated* adalah simpul yang tidak incident dengan setiap sisi pada matching M. (Gondran dan Minoux, 1984: 280)

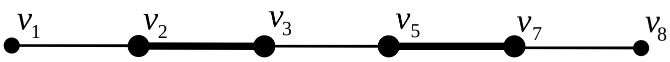
   Contoh : simpul v_8 dan v_1

Selain simpul terdapat dua istilah lintasan pada *matching*, yakni:

1. ***Lintasan Alternating***. Lintasan *alternating* adalah lintasan pada graf G yang sisi-sisinya bergantian antara matching dan bukan matching. (Gondran dan Minoux, 1984: 281). Contoh: lintasan $v_8 - v_7 - v_5 - v_4 - v_6$



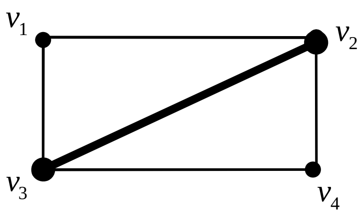
2. ***Lintasan Augmenting***. Lintasan *augmenting* adalah lintasan *alternating* yang simpul pangkal dan simpul ujungnya adalah simpul *unsaturated* (Gondran dan Minoux, 1984: 281). Contoh : lintasan $v_1 - v_2 - v_3 - v -_5 v_7 - v_8$



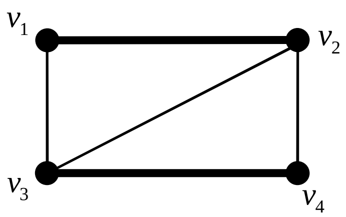
Matching Maksimum

M disebut *matching* maksimum jika G tidak memuat *matching* M' yang lain dengan $|M'| > |M|$. (Bondy dan Murty, 1976: 70)

Contoh:



Graf dengan *matching* yang belum maksimum



Graf dengan *matching* yang maksimum

Pohon Alternating

Pohon adalah sebuah graf terhubung yang mempunyai n buah simpul, (n-1) sisi dan tidak mempunyai *cycle* (Samuel, 2008: 159). Sedangkan pohon *alternating T* merupakan pohon berakar di simpul *v*, jika setiap lintasan yang berawal di *v* adalah lintasan *alternating-M* (Chartrand dan Oellerman, 1993:169).

Terdapat 2 bentuk pohon alternating dengan akar *v*, yakni:

1. Semua simpul pada T kecuali *v* adalah simpul *saturated-M*.
2. Pohon T memuat sebuah simpul *unsaturated-M* selain simpul *v*

Masalah Penugasan

Masalah penugasan dapat dikelompokkan menjadi dua, yaitu :

1. Masalah minimisasi, yaitu masalah penugasan yang mencari total kerugian minimum.
2. Masalah maksimasi, yaitu masalah penugasan yang mencari total keuntungan maksimum.

Penyelesaian masalah penugasan sama artinya dengan menentukan 1-*regular subgraf* pada graf bipartit G dengan kardinalitas maksimum. (Chatrand dan Oellerman, 1993: 162)

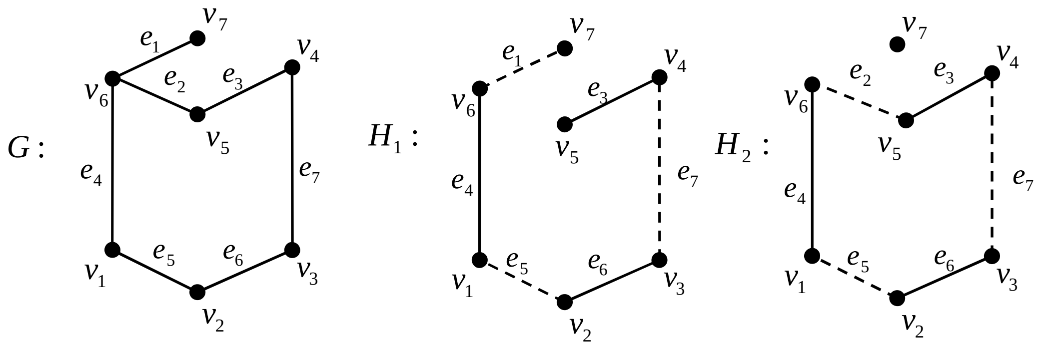
Pada kondisi dimana penilaian menjadi pertimbangan, maka hal ini dapat diselesaikan dengan menentukan sebuah 1-*regular subgraf H* pada graf berbobot G dengan penjumlahan bobot sisi-sisi pada H sehingga G juga maksimum. (Chatrand dan Oellerman, 1993: 162)

Penyelesaian Matching Maksimum dengan Metode Hungarian

Teorema-teorema yang menjadi dasar dalam penentuan *matching* maksimum pada graf bipartit berbobot menggunakan metode Hungarian**. (Berge, 1984: 122)**

Misal M_1 dan M_2 adalah matching di graf G. Jika H adalah subgraf perentang dari G dengan himpunan sisi $E(H) = M_1 \otimes M_2 = (M_1 - M_2) \cup (M_2 - M_1)$ maka setiap komponen dari H akan mengikuti salah satu dari tipe berikut:

1. Sebuah simpul terasing.
2. Sebuah cycle genap dengan sisi bergantian antara M_1 dan M_2
3. Sebuah lintasan dengan sisi saling bergantian *antara* M_1 dan M_2, sedemikian sehingga setiap simpul terakhir dari lintasan adalah simpul *unsaturated* di M_1 dan M_2, tetapi bukan di keduanya. Ilustrasinya sebagai berikut:

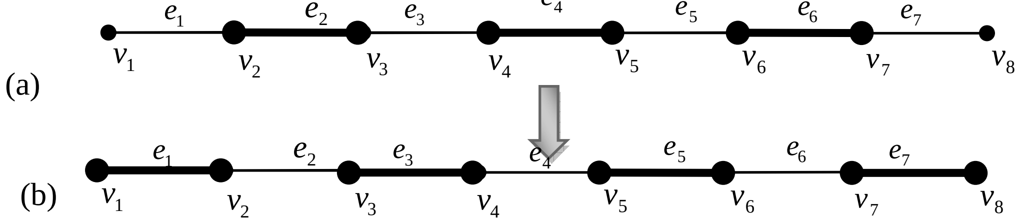


H_1 dan H_2 adalah subgraf perentang G

Teorema: (Bondy dan Murty, 1976: 70)

Matching M di G adalah matching *maksimum jika dan hanya jika G tidak memuat lintasan augmenting.* Sebagai ilustrasi dari teorema ini diberikan graf *G* pada Gambar di bawah ini. Misalkan $M' = \{e_2, e_4, e_6\}$ adalah *matching* di *G* dan $P: v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8$ merupakan lintasan *augmenting-M* maka $M = \{e_1, e_3, e_5, e_7\}$ adalah *matching* yang diperoleh dengan *augmenting-M* sepanjang *P*, dengan $|M| = 4 = |M'| + 1$

Contoh : *Augmenting-M* sepanjang *P*



Matching Maksimum pada Graf Bipartit

Matching maksimum pada graf bipartit adalah banyaknya *matching* yang termuat pada suatu graf bipartit dengan kardinalitas yang maksimum.

Misalkan *G* merupakan graf bipartit dengan partisi (X,Y). Jika $S \subseteq X$ maka persekitaran *S* adalah $N(S)$ yaitu simpul-simpul yang *adjacent* dengan simpul di *S*.

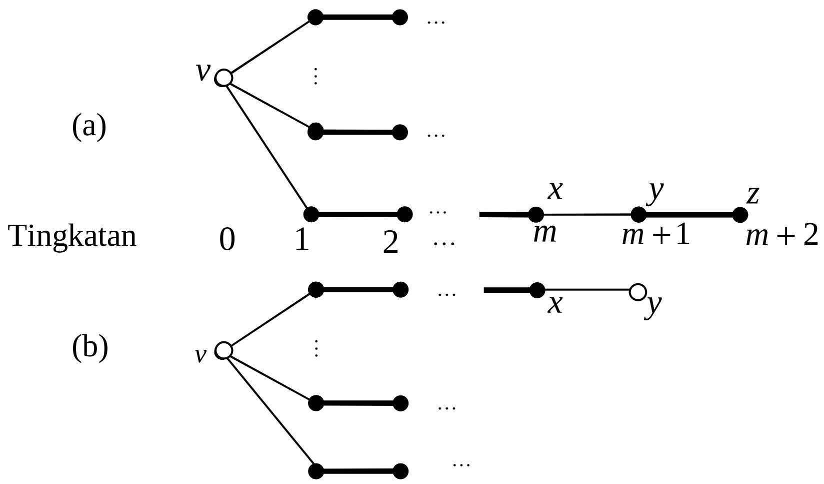
Teorema (Kocay dan Kreher, 2004: 141)

Misalkan *G adalah graf bipartit dengan partisi* (X,Y) *maka G memuat sebuah matching yang memenuhi setiap simpul* X *jika dan hanya jika* $|N(S)| \geq |S|$ *untuk setiap* $S \subseteq X$ **.**

Teorema ini membantu dalam penyusunan pohon *alternating*. Chartrand dan Oellermann (1993: 163) memberikan cara menyusun pohon *alternating* yang berakar pada v, yaitu dengan menempatkan *v* pada tingkatan 0 dan semua simpul $u_1, u_2, \ldots, u_k$ yang *adjacent* dengan *v* di *G* ditempatkan pada tingkatan 1, dan simpul *v* dihubungkan pada simpul $u_i(1 \leq i \leq k)$. Misal $u_1 v_i$ untuk $1 \leq i \leq k$. Simpul $v_1, v_2, \ldots, v_k$ ditempatkan pada tingkatan 2, dan simpul *u* dihubungkan pada simpul $v_i(1 \leq i \leq k)$. Jika pohon *alternating* sudah tersusun pada tingkatan genap (*m* genap) maka setiap simpul ada tingkat *m* digunakan untuk menguji simpul y *adjacent*.

Simpul y adalah simpul di $Y - T$ yang *adjacent* pada sebuah simpul x di S. Jika y adalah simpul *saturated* dengan $yz \in M$ dimana z bukan termasuk pohon *alternating*, sehingga simpul y ditempatkan pada tingkatan $m+1$ dan z ditempatkan pada $m+2$, dengan demikian *adjacent* dengan y dan y *adjacent* dengan z, dan akan terbentuk pohon *alternating* seperti gambar (a) di bawah ini. Jika y adalah simpul *unsaturated*, maka pohon *alternating* dibangun dengan menambahkan simpul y dan sisi xy, sehingga didapat lintasan *augmenting* dari simpul v ke simpul y seperti gambar (b) di bawah ini. Dengan lintasan *augmenting* ini dapat dibentuk *matching* baru yang lebih besar.

Contoh proses membangun pohon *alternating*



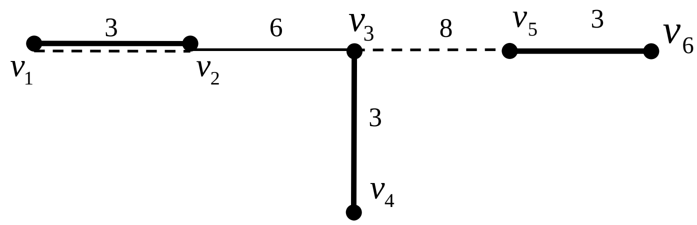
Susunan pohon *alternating* berhenti jika tidak bisa menambah tingkatan pohon *alternating* atau lintasan *augmenting* telah ditemukan.

Setiap k-regular graf bipartit mempunyai matching sempurna jika k > 0. Matching sempurna pada graf bipartit menghasilkan kardinalitasnya sebanyak $\left|\frac{|V(G)|}{2}\right|$ (Kocay dan Kreher, 2004: 143)

Matching Maksimum pada Graf Berbobot

Matching maksimum pada graf berbobot merupakan *matching* yang memiliki jumlah bobot sisi yang maksimum**.** (Chartrand dan Oellermann, 1993: 163). *Matching* dengan bobot maksimum belum tentu memiliki kardinalitas maksimum. Contoh *Matching* berbobot maksimum dan *matching* maksimum:

.



Matching Maksimum pada Graf Bipartit Berbobot

Matching maksimum pada graf bipartit berbobot merupakan *matching* sempurna dengan jumlah bobot sisi yang maksimum pada graf bipartit berbobot.

Misalkan G adalah graf bipartit berbobot dengan partisi himpunan V_1 dan V_2. Graf G' adalah graf bipartit lengkap berbobot yang memuat G sebagai subgraf. Misalkan G' memiliki partisi himpunan U_1 dan U_2 dengan $|U_1| = |U_2| =$ maksimum $\{|V_1|,|V_2|\}$ dan V_i termuat di U_i , untuk $i = 1,2$. Misal $x \in U_1$ dan $y \in U_2$, jika $w_{G'}(xy) = w_G(xy)$ maka $xy \in E(G)$ dan jika $w_{G'}(xy) = 0$, maka $xy \in E(G)$.

Jika M adalah *matching* sempurna dengan jumlah bobot maksimum di G', maka $M \cap E(G)$ adalah *matching* maksimum di G dengan kardinalitas $\left|\frac{|V(G)|}{2}\right|$.

Pelabelan simpul atau $\ell(x)$ adalah fungsi nilai real ℓ dari himpunan simpul $V_1 \cup V_2$ untuk setiap $v \in V_1$ dan $u \in V_2$; $\ell(v) + \ell(u) \geq w(v,u)$

dimisalkan $\ell(v) = \max_{u \in V_2} w(v,u)$ untuk setiap $v \in V_1$

$\ell(u) = 0$ untuk setiap $u \in V_2$

maka adalah pelabelan simpul di G' dengan

$E_\ell = \{vuE(G') | v \in V_1 \text{ dan } u \in V_2 \text{ dan } \ell(v) + \ell(u) = w(vu)\}$

Teorema (Chartrand dan Oellermann, 1993: 174)

Jika ℓ *adalah pelabelan simpul pada graf bipartit lengkap berbobot* G'. *Misal* H_ℓ *adalah subgraf perentang dari* G'. *Jika* E_ℓ *memuat matching sempurna* M', *maka* M' *merupakan matching berbobot maksimum di* G'

Metode Hungarian

Berikut adalah metode hungarian untuk menentukan matching maksimum pada graf biparti berbobot. Misalkan G' adalah graf bipartit berbobot dengan partisi himpunan simpul V_1 dan V_2.

1. Langkah pertama adalah melakukan pelabelan simpul ℓ
   1.1. Untuk $\forall\, v \in V_1$, misal $\ell(v) = \max\limits_{u \in V_2} w(v,u)$
   1.2. Untuk $\forall\, u \in V_2$, misal $\ell(v) = 0$
   1.3. H_l adalah subgraf perentang dari G_l dengan himpunan sisi E_l.
   1.4. Misal G_l adalah graf dasar dari H_l.
2. Pilih sembarang *matching M* di G_ℓ
3. Jika mendapatkan *matching* maksimum di G_ℓ, dicari apakah G_ℓ merupakan *matching* sempurna maka menurut teorema di atas *matching* sempurna tersebut adalah *matching* maksimum di G'.
   Jika *matching* maksimum di G_ℓ tersebut bukan *matching* sempurna, maka susun sebuah pohon *alternating T* yang berakar di simpul *unsaturated* untuk menentukan pelabelan simpul baru ℓ'.
   Berikut merupakan langkah-langkah untuk menentukan sebuah pelabelan simpul baru ℓ'.
   3.1. Jika setiap simpul di V_1 merupakan simpul saturated di M, maka M adalah matching maksimum di G' dan proses berhenti. Jika tidak maka lanjutkan.
   3.2. Misal x adalah simpul unsaturated di V_1.
   3.3. Susun pohon alternating dari M yang berakar di x.
   Jika terdapat lintasan augmenting-P di G_ℓ, maka gantikan M dengan $M' = M \otimes E(P)$ untuk mendapatkan matching baru.
   Jika tidak memuat lintasan augmenting P, dan T adalah pohon alternating dari M dengan akar x yang tidak dapat diperluas lagi di G_ℓ, maka pelabelan simpul ℓ diganti dengan sebuah pelabelan simpul baru ℓ' dengan sifat bahwa M dan T termuat di graf dasar G_ℓ dari $H_{\ell'}$.
4. Menghitung pelabelan simpul baru ℓ'
   Misalkan:

$$m_\ell = \text{minimal } \{\ell(v) + \ell(u) - w(vu) \mid v \in V_1 \cap V(T) \text{ dan } u \in V_2 - V(T)\}$$

Maka berlaku untuk ℓ':

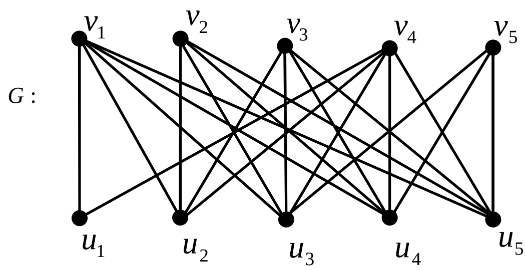
$$\ell'(v) = \begin{cases} \ell(v) - m\ell & untuk \quad v \in V_1 \cap V(T) \\ \ell(v) + m\ell & untuk \quad v \in V_2 \cap V(T) \\ \ell(v) & yang \quad lainnya \end{cases}$$

5. Jika pelabelan simpul yang baru belum memuat semua simpul di V_1, maka ulang ke langkah 3.3.

Contoh Kasus

Misal seorang manager sebuah perusahaan melakukan seleksi terdapat 5 calon pegawai $v_i\,(i = 1,2,\dots,5)$ untuk 5 posisi jabatan $u_i\,(i = 1,2,\dots,5)$. Seleksi dilakukan untuk mengetahui kemampuan setiap pelamar untuk setiap posisi. Kemampuan pelamar kerja merupakan bobot sisi yang menghubungkan pelamar kerja dengan jabatan. Bentuk matrik $M = \lfloor m_{ij} \rfloor$ dengan $m_{ij} = w(v_i u_j)$

$$M = \begin{array}{cccccc} & u_1 & u_2 & u_3 & u_4 & u_5 \\ v_1 & 4 & 3 & 1 & 9 & 2 \\ v_2 & 0 & 3 & 2 & 4 & 2 \\ v_3 & 0 & 3 & 4 & 6 & 4 \\ v_4 & 1 & 1 & 2 & 2 & 2 \\ v_5 & 0 & 0 & 3 & 3 & 3 \end{array}$$



G adalah graf bipartit berbobot dengan partisi himpunan simpul dengan partisi himpunan $V_1 = (v_1, v_2, v_3, v_4, v_5)$ dan $V_2 = (u_1, u_2, u_3, u_4, u_5)$ dengan mengikuti langkah-langkah pada metode Hungarian diperoleh hasil sebagai berikut:

Langkah 1.a Untuk $\forall\, v \in V_1$, misal $\ell(v) = \max_{u \in V_2} w(v,u)$

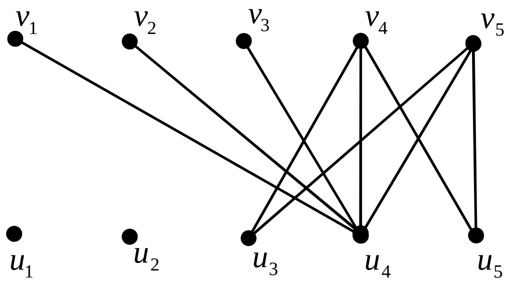
Sehingga diperoleh $\ell(v) = (9,4,6,2,3)$

Untuk $\forall\, u \in V_2$, misal $\ell(u) = 0$

maka nilai pelabelan $\ell(u) = (0,0,0,0,0)$

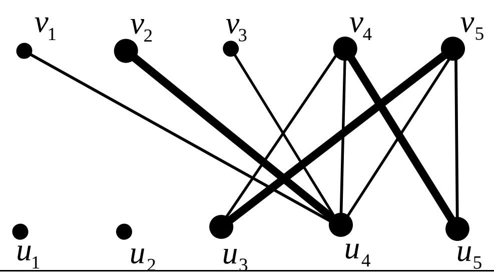
1.b Misal H_ℓ adalah subgraf perentang dari G' dengan himpunan sisi E_ℓ

1.c Misal G_ℓ adalah graf dasar dari H_ℓ



Langkah 2. Pilih sembarang *matching M* di G_ℓ

Misal dipilih $M_0 = (v_2, u_4), (v_4, u_5), (v_5, u_3)$

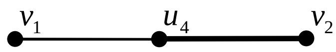


Langkah 3.a Terdapat simpul pada V_1 yang merupakan simpul *unsaturated* di *M*, maka lanjutkan.

3.b v_1 adalah simpul *unsaturated* di V_1 sehingga v_1 digunakan sebagai simpul akar.

3.c Susun pohon *alternating* dari *M* yang berakar di v_1. Diperoleh pohon *alternating* T dengan $V(T)=\{v_1,u_4,v_2\}$, karena tidak ditemukan lintasan *augmenting* dan pohon *alternating* T tidak dapat diperluas lagi di G_ℓ, maka pelabelan simpul ℓ diganti dengan sebuah pelabelan simpul baru ℓ'.

Pohon *alternating* pertama



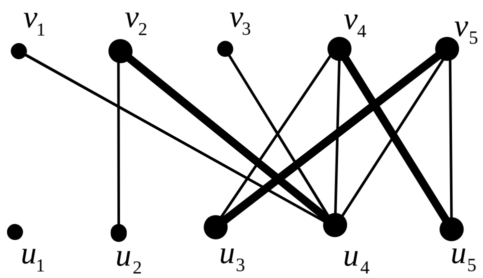
Langkah 4. Menghitung pelabelan baru ℓ'

$m_\ell =$ minimal $\{\ell(v)+\ell(u)-w(vu) \mid v\in V_1 \cap V(T)$ dan $u \in V_2 - V(T)\}$

$v\in V_1 \cap V(T)=\{v_1,v_2\}$ dan $u\in V_2 - V(T)=\{u_1,u_2,u_3,u_5\}$

diperoleh $m_\ell = 1$ yakni pada sisi (v_2u_2) sehingga pelabelan simpul baru $\ell'(v)$ yang terbentuk adalah:

$$\ell'(v)=\begin{cases} \ell(v)-m\ell=\{8,3,6,2,3\} & untuk \quad v\in V_1 \cap V(T) \\ \ell(u)+m\ell=\{0,0,0,1,0\} & untuk \quad u\in V_2 \cap V(T) \\ \ell(v) & untuk \quad yang \quad lainnya \end{cases}$$



Matching M_0 dan sisi m_ℓ

Langkah 5. Karena pelabelan simpul ℓ' yang baru belum memuat semua simpul ℓ, maka ulang ke langkah 3.c

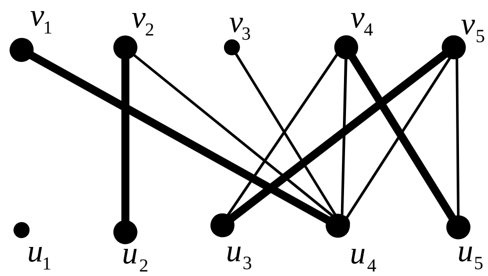
Langkah 3.c Susun pohon *alternating* dari *M* yang berakar di v_1. Diperoleh pohon *alternating* T dengan $V(T)=\{v_1,u_4,v_2,u_2\}$, karena ditemukan lintasan *augmenting-M* yakni $P_0=\{(v_1,u_4),(v_2,u_4),(v_2,u_2)\}$ maka lintasan *augmenting* P_0 digunakan untuk membentuk *matching* baru dan kembali ke langkah 3.a.

Pohon *alternating* kedua

$M_0 = \{(v_2, u_4), (v_4, u_5), (v_5, u_3)\}$

$P_0 = \{(v_1, u_4), (v_2, u_4), (v_2, u_2)\}$

maka $M_1 = M_0 \otimes P_0 = \{(v_4, u_5), (v_5, u_3)(v_1, u_4), (v_2, u_2)\}$

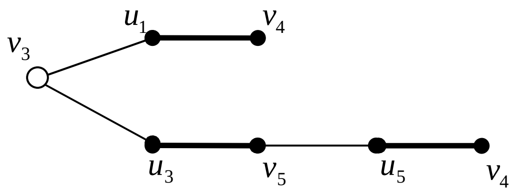


Matching M_1

Langkah 3.a Terdapat simpul pada V_1 yang merupakan simpul *unsaturated* di *M*, maka lanjutkan.

Langkah 3.b v_3 adalah simpul *unsaturated* di V_1 sehingga v_3 digunakan sebagai simpul akar.

Langkah 3.c Susun pohon *alternating* dari *M* yang berakar di v_3. Diperoleh pohon *alternating* *T* dengan $V(T) = \{v_3, u_4, v_1, u_3, v_5, u_5, v_4\}$, karena tidak ditemukan lintasan *augmenting*, dan pohon *alternating* *T* tidak dapat diperluas lagi di G_ℓ, maka pelabelan simpul ℓ diganti dengan sebuah pelabelan simpul baru ℓ'



Pohon *alternating* ketiga

Langkah 4. Menghitung pelabelan baru ℓ'

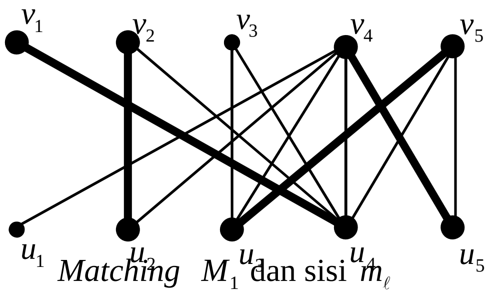
$m_\ell =$ minimal $\{\ell(v) + \ell(u) - w(vu) \mid v \in V_1 \cap V(T)$ dan $u \in V_2 - V(T)\}$

$v \in V_1 \cap V(T) = \{v_1, v_3, v_4, v_5\}$ dan $u \in V_2 - V(T) = \{u_1, u_2\}$

dari perhitungan diperoleh $m_\ell = 1$ yakni pada sisi $(v_4 u_1)$ dan sisi $(v_4 u_2)$, pelabelan baru yang terbentuk adalah :

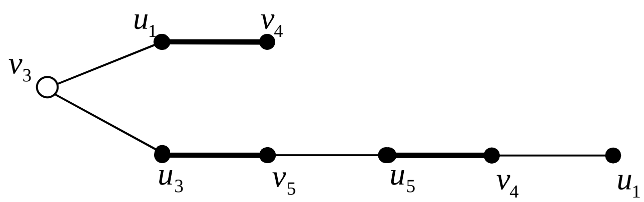
$$\ell'(v)=\begin{cases}\ell(v)-m\ell=\{7,3,5,1,2\} & untuk \quad v\in V_1\cap V(T)\\ \ell(u)+m\ell=\{0,0,1,2,1\} & untuk \quad u\in V_2\cap V(T)\\ \ell(v) & untuk \quad yang \quad lainnya\end{cases}$$

Langkah 5. Karena pelabelan simpul ℓ' yang baru belum memuat semua simpul ℓ, maka ulang ke langkah 3.c.



Matching M_1 dan sisi m_ℓ

Langkah 3.c Susun pohon *alternating* dari *M* yang berakar di v_3. Diperoleh pohon *alternating* *T* dengan $V(T)=\{v_3,u_4,v_1,u_3,v_5,u_5,v_4,u_1\}$, karena diperoleh lintasan *augmenting* $P_1=\{(v_3,u_3),(v_5,u_3),(v_5,u_5),(v_4,u_5),(v_4,u_1)\}$ maka lintasan *augmenting* P_1 digunakan untuk membentuk *matching* baru dan kembali pada langkah 3.a

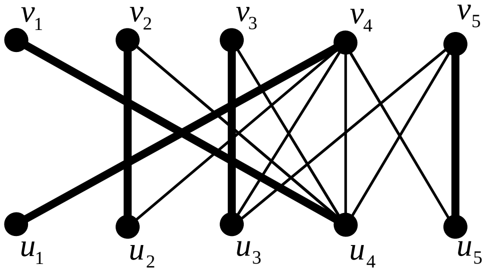


Pohon *alternating* keempat

$$M_1=\{(v_4,u_5),(v_5,u_3)(v_1,u_4),(v_2,u_2)\}$$

$$P_1=\{(v_3,u_3),(v_5,u_3),(v_5,u_5),(v_4,u_5),(v_4,u_1)\}$$

maka $M_2=M_1\otimes P_1=\{(v_3,u_3),(v_2,u_2),(v_5,u_5),(v_1,u_4),(v_4,u_1)\}$



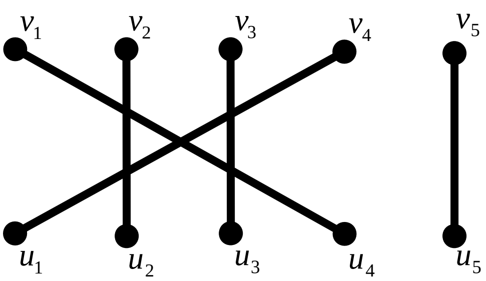
Matching M_2

$$
M = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ \ell(u_i) \end{array}
\begin{array}{c}
\begin{array}{ccccc} u_1 & u_2 & u_3 & u_4 & u_5 \end{array} \\
\begin{bmatrix} 4 & 3 & 1 & 9 & 2 \\ 0 & 3 & 2 & 4 & 2 \\ 0 & 3 & 4 & 6 & 4 \\ 1 & 1 & 2 & 2 & 2 \\ 0 & 0 & 3 & 3 & 3 \end{bmatrix} \\
\begin{array}{ccccc} 0 & 0 & 1 & 2 & 1 \end{array}
\end{array}
\begin{array}{c} \ell(v_i) \\ 7 \\ 3 \\ 5 \\ 1 \\ 2 \\ \\ \end{array}
$$

Langkah 3.a Setiap simpul di V_1 merupakan simpul *saturated* di M_2 maka *matching* adalah *matching* maksimum di G' dan proses berhenti.
Jadi $M_2 \cap E(G) = \{(v_3,u_3),(v_2,u_2),(v_5,u_5),(v_1,u_4),(v_4,u_1)\}$ adalah *matching* maksimum di *G*.

Dari langkah-langkah di atas diperoleh *matching* sempurna dengan bobot maksimum sebesar 20. dan kardinalitas *matching M* adalah 5. Jadi penempatan calon pegawai pada posisi jabatan adalah sebagai berikut:

calon pegawai v_1 pada jabatan u_4,
calon pegawai v_2 pada jabatan v_2,
calon pegawai v_3 pada jabatan u_3,
calon pegawai v_4 pada jabatan u_1, dan,
calon pegawai v_5 pada jabatan u_5.



Solusi penempatan pegawai

E. Kesimpulan

Matching maksimum pada graf bipartit berbobot dapat ditentukan menggunakan metode Hungarian. *Matching* tersebut merupakan *matching* sempurna dengan dengan kardinalitas dan jumlah bobot sisi yang maksimum pada graf bipartit berbobot.

Matching maksimum yang diperoleh pada graf bipartit berbobot merupakan solusi optimal dari masalah penugasan yakni memasangkan seorang pegawai dengan sebuah tugas**.**

Metode Hungarian dapat membantu perusahaan atau institusi dalam mengambil keputusan terutama menyangkut pemberian pekerjaan, penempatan posisi pegawai, pembuatan jadwal dan lain sebagainya.

Daftar Pustaka

Anton, Howard, 1987, *Aljabar Linear* Elementer: Jakarta.

Berge, Coulde, 1970, *Graph and Hypergraph*, Dunod: Netherlands.

Bondy, J.A dan Murty, U.S.R., 1976, *Graph Theory with Applications*, Mac Millan Press: New York.

Chartrand, Gary dan Oellermann, Ortrud, 1993, *Applied and Algoritmic Graph Theory*, Mc Graw Hill International Edition: New York.

Gondran, M. dan Minoux, M., 1984, *Graphs and Algorithms*. John Wiley & Sons, Ltd.: Chichester, Inggris.

Kocay, William dan Kreher, Donald, 2005, *Graph Theory and Optimization*, Chapman & Hall / CRC.

Magun, J., 2000. *Greedy Matching Algorithms, an Experimental Study*. Schweizerischer Nationalfond, Grant NF 2000-422 44.94. Zűrich, Switzerland.

Venkateswaran, R., Obranivic, Z., Raghavendra, C.S., 1993. *Cooperative Genetic for Optimization Problem in Distributed Computer System*. Technical report TR-EECS-93-018. School of EECS. Washington State University.

Wibisono, Samuel, 2008, *Matematika Diskrit*. Edisi Kedua, Graha Ilmu: Yogyakarta.