Optimization Problems for Blockers and Transversals in a Graph

D. de Werra (EPFL,Lausanne)

Shenyang
October 2012

Optimization Problems for Blockers and Transversals in a Graph

D. de Werra (EPFL,Lausanne)
Joint work with
C.Bentz (Uni Paris Sud)
M.C. Costa (ENSTA, Paris)
C. Picouleau (CNAM, Paris)
B.Ries (Uni Dauphine, Paris)
R.Zenklusen (MIT)

- Find most vital elements of a system
  (for protection)

Player A chooses any optimal action plan

Player P wants to prevent this

Example: $G = (V,E)$ graph

$\mathcal{C}(V) = (S \mid S \subseteq V$ stable set in $G)$

$\alpha(G) = \max(|S|\ S$: stable)

Player A: Find a maximum stable set S in G

(set of possible « independent » stores to open)

Example: $G = (V,E)$ graph

$\mathcal{C}(V) = (S \mid S \subseteq V$ stable set in $G)$

$\alpha(G) = \max|S|\ S$: stable)

Player A: Find a maximum stable set S in G

(set of possible « independent » stores to open)

Player P: Find a minimum $T \subseteq V$

with $|S \cap T| \geq d \qquad \forall S$ stable with $|S| = \alpha(G)$

« *d-transversal* »

Example: $G = (V,E)$ graph

$\mathcal{C}(V) = (S \mid S \subseteq V$ stable set in $G)$

$\alpha(G) = \max|S|\ S$: stable)

Player A: Find a maximum stable set S in G

(set of possible « independent » stores to open)

Player P: Find a minimum $T \subseteq V$

with $|S \cap T| \geq d \quad \forall S$ stable with $|S| = \alpha(G)$

« *d-transversal* »

After removal of T all maximum action plans have lost at least *d stores*

Another option for P:

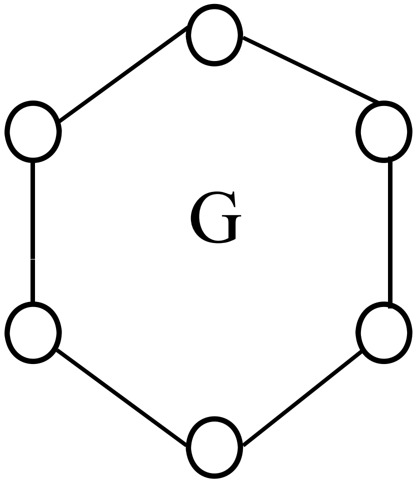
Find smallest subset B of V such that after removal of B no action plan (maximum set of « independent » stores) has more than

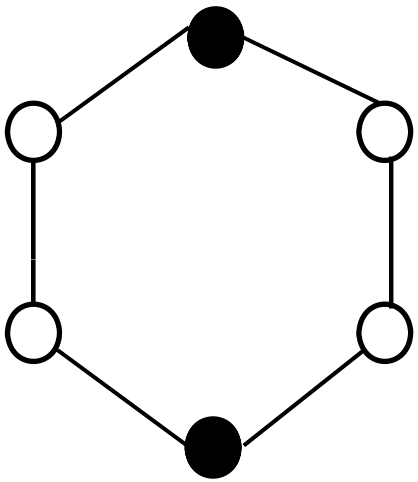
$\alpha(G)-d$ stores.

Player P: Find a minimum $B \subseteq V$ with $\alpha\big(G'=(V-B,E')\big)\leq \alpha(G)-d$

« d-blocker »

Illustration: $T = \{\bullet\} = B$

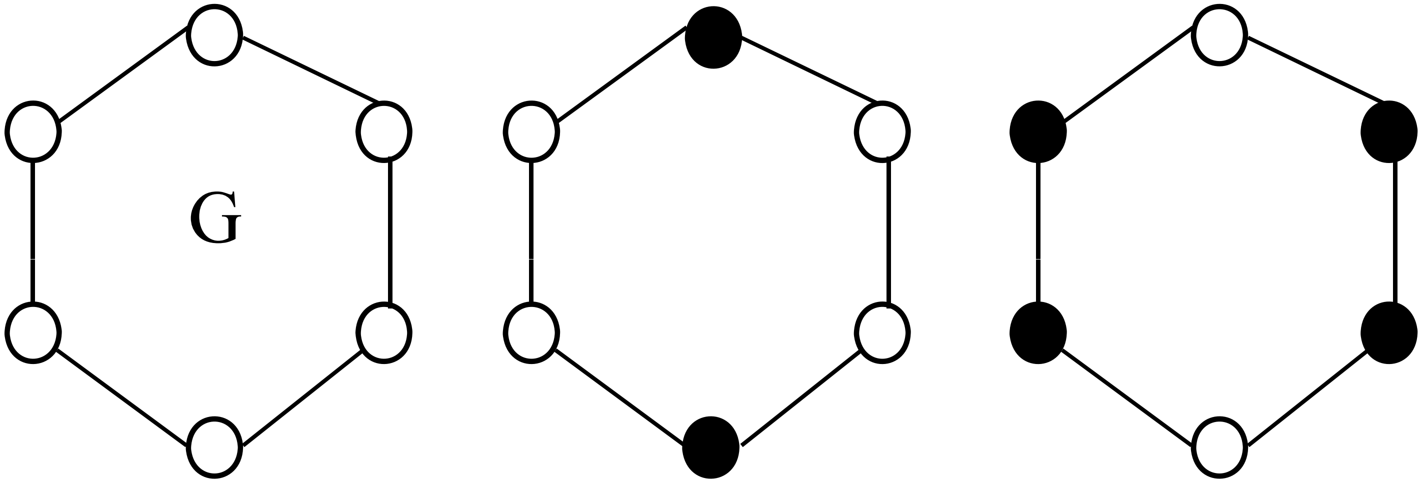




$\alpha(G) = 3$ $d = 1$

Illustration: $T = \{\bullet\} = B$ $T = \{\bullet\}$

2-transversal



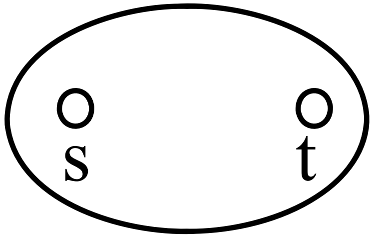
NOT

2-blocker!

$\alpha(G) = 3$ $d = 1$ $d = 2$

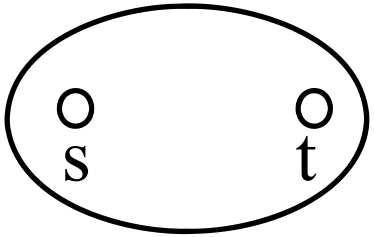
$\alpha(G)\text{-}d = 1$

Other examples: shortest *s*-*t* paths



every arc is in a shortest *s*-*t* path

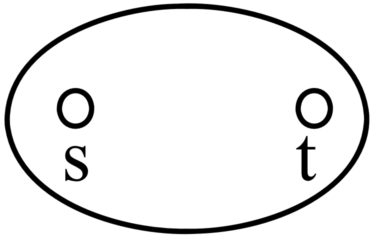
Other examples: shortest s-t paths



every arc is in a shortest s-t path

T is an (inclusionwise) minimal d-transversal $\Leftrightarrow$ $T = d$ disjoint s-t cuts

Other examples: shortest *s*-*t* paths



every arc is in a shortest
s-*t* path

T is an (inclusionwise) minimal
d-transversal $\Leftrightarrow T = d$ disjoint
s-*t* cuts

Finding a minimum d-transversal
$\leftrightarrow$ finding disjoint *s*-*t* cuts $C_1, \ldots, C_d$

with $|C_1| + \ldots + |C_d|$ minimum.

$\exists$ polynomial algorithm (D. Wagner, 90)

d-blocker B: subset of arcs whose removal increases $\ell(G=(V,E))$

by at least d

where $\ell\left(G=\left(V,E\right)\right)$ = length of shortest s-t path in G

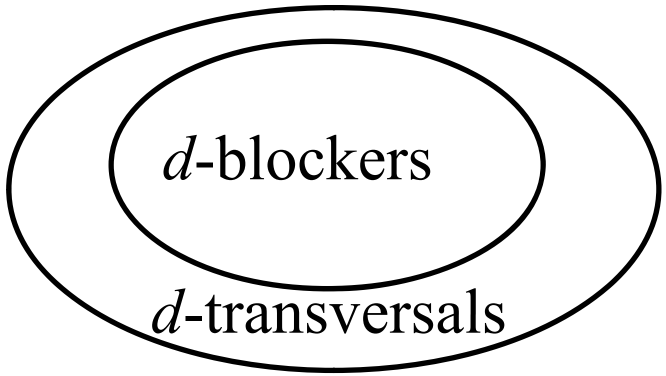
we have $\ell\left(G'=\left(V,E-B\right)\right)\geq\ \ell\left(G=\left(V,E\right)\right)+d$

NP-hard (Khachiyan et al, 08)

Fact 1 : A subset $T \subseteq V$ is a *1*-transversal

$\Leftrightarrow$ T is a *1*-blocker

Fact 2 : For $d > 1$

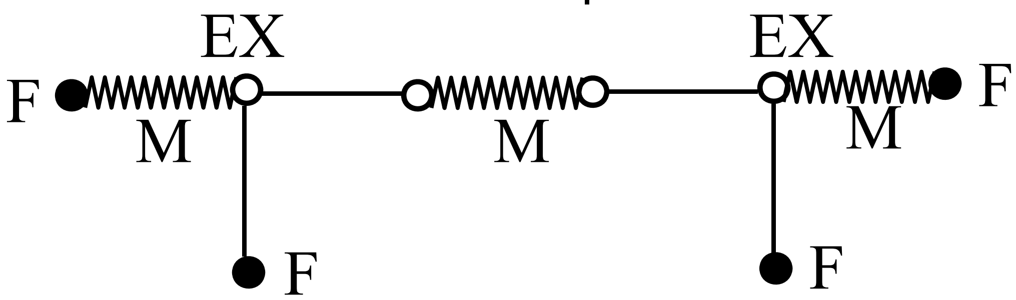


$\mathcal{C}$: Stable sets in bipartite graphs $\boldsymbol{G=(V,E)}$

v is **forced** if $v \in S$

v is **excluded** if $v \notin S$

$\forall$ S maximum stable set



More General Formulation

Given finite ground set V, integer d

$$\mathcal{C}(V) = (C \mid C \subseteq V, C \text{ has property } P^*)$$

(ex. P* : S is a stable set in G)

Each element v in V has a **weight** $w(v)$ and a **cost** $c(v)$

$$h(S) = \sum (h(v) \mid v \in S) \quad \text{for } h = w, c$$

Player A: Find a $C \in \mathcal{C}$ with maximum weight $w(C)$

$$\alpha_w\big(\mathcal{C}(V)\big) = \max\left\{ w(C) \middle| C \in \mathcal{C} \right\}$$

Player P: Find a subset $T \subseteq V$ with minimum cost $c(T)$ such that $|T \cap C| \geq d$ for all max weight C in $\mathcal{C}$

«generalized d-transversal »

Given finite ground set V, integer d

$$\mathcal{C}(V) = (C \mid C \subseteq V, C \text{ has property } P)$$

Player A: Find a $C \in \mathcal{C}$ with maximum weight $w(C)$

Player P: Find a subset $B \subseteq V$ with minimum cost $c(B)$
such that $\alpha_w\big(\mathcal{C}(V-B)\big) \le \alpha_w\big(\mathcal{C}(V)\big) - d$

« generalized d-blocker »

GENTRANS

Find a generalized d-transversal T (of the maximum weight subsets C in $\mathcal{C}$) with minimum cost $c(T)$

GENBLOCK

Find a generalized d-blocker B (of the maximum weight subsets C in $\mathcal{C}$) with minimum cost $c(B)$

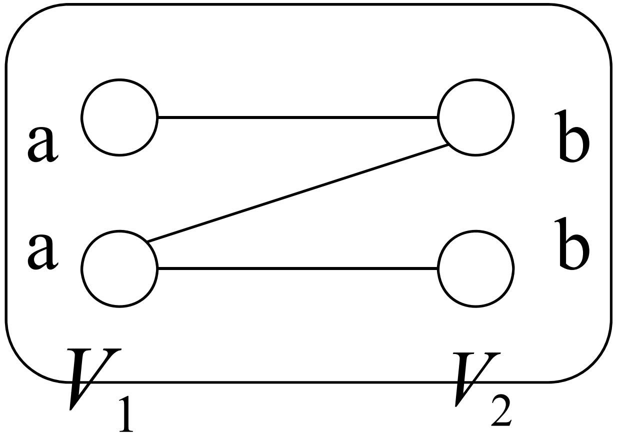
NB if $w(v)=c(v)=1$ d-transversals and d-blockers

GENTRANS($\mathcal{C}$, w, $c=1$, $i=w$, d) polynomially solvable if G is cobipartite

GENTRANS($\mathcal{C}$, *w*, *c=1*, *i=w*, *d*) polynomially solvable if G is cobipartite

$$|S| \leq 2 \qquad 1 \leq d \leq \alpha_w(G)$$

Introduce all vertices *v* with $w(v) = \alpha_w(G)$ into T



$$a + b = \alpha_w(G)$$

$a \geq b \geq d$ min VC of H into T

$a \geq d > b$ V_1 into T

$d > a \geq b$ V into T

connected component of $\overline{\mathrm{G}}$

GENBLOCK($\mathcal{C}$, *w*, *c=1*, *d*) is polynomially solvable if G is cobipartite

DGENBLOCK($\mathcal{C}$,w,c,d,k) : Is there a generalized d-blocker B of all max weight sets in $\mathcal{C}$ with $c(B) \leq k$?

NB: As shown before

DGENBLOCK($\mathcal{C}$,$w=1,c=1, d,k=d$) is NP-complete if G is a split graph (2010)

DGENBLOCK($\mathcal{C}$,$w=1,c=1,d,k$) and

DGENTRANS($\mathcal{C}$,$w=1,c=1, d,k$) are NP-complete if G is line graph of a bipartite graph (2009)

DGENBLOCK($\mathcal{C}$,w,$c=1$, d,k) is NP-complete if G is a bipartite graph (S.Toubaline, 2010)

GENBLOCK($\mathcal{C}$,w,$c=1$,d,k) is polynomially solvable if G is a tree or a cograph

(S.Toubaline, 2010)

GENBLOCK($\mathcal{C}$,*w=1,c=1,d,*) and GENTRANS($\mathcal{C}$,*w=1,c=1,d,)* can be solved in polynomial time if G is a grid graph (2010).

GENTRANS($\mathcal{C}, w, c, d)$ is polynomially solvable if G is bipartite

Minimum cost d-transversal of maximum weight stable sets in a bipartite graph

Minimum cost d-transversal of maximum weight stable sets in a bipartite graph

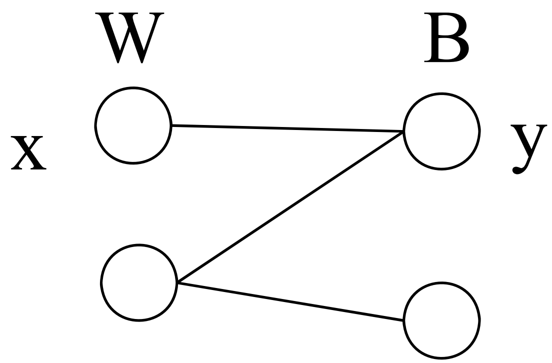
Remove all forbidden vertices

Discard all forced vertices

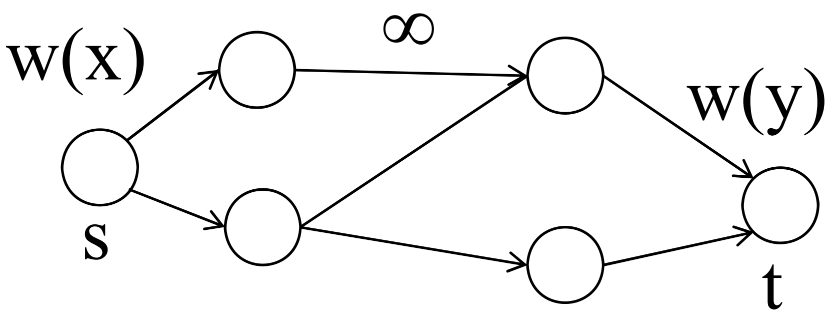
In remaining graph all vertices are free

G = (B, W, E)   B= black vertices

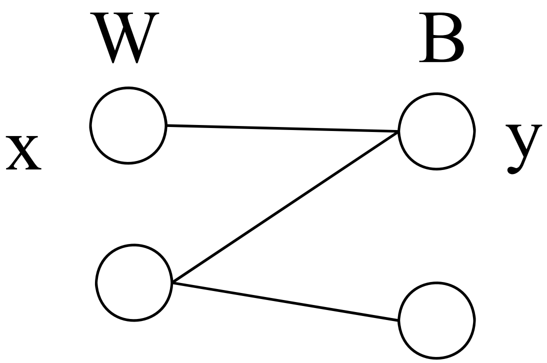
W= white vertices



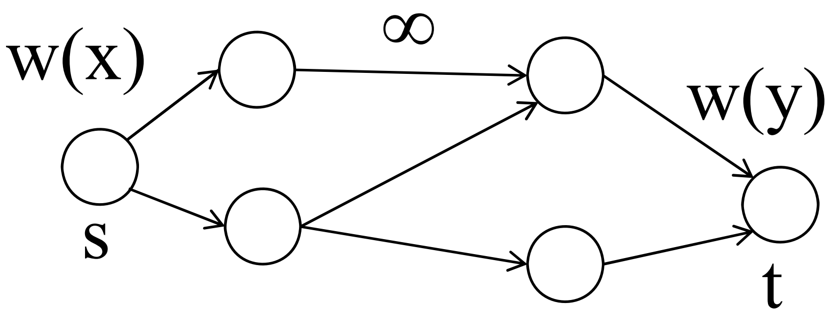
graph G with weights



network N with
capacities



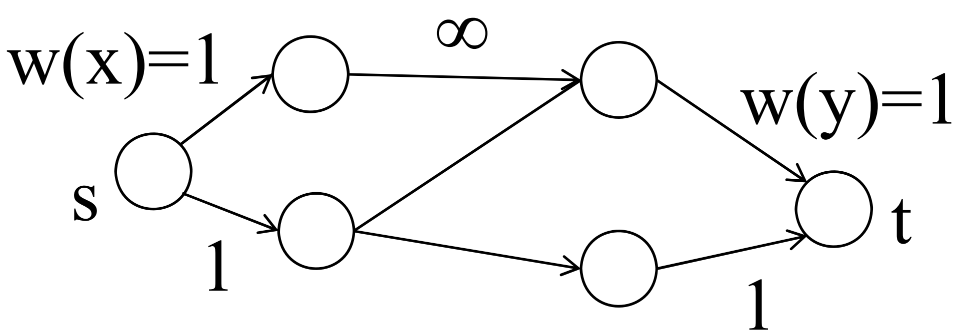
graph G with weights



network N with capacities

In G all vertices are free if and only if in N there is a flow from s to t with value

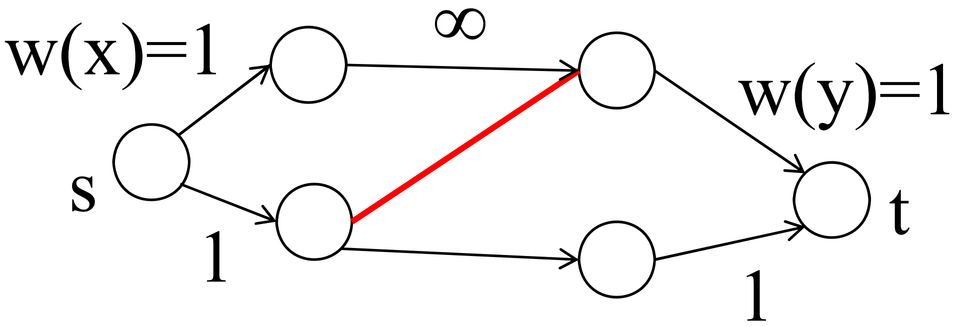
$$w(W) = w(B)$$



network N with
capacities

F* = forbidden arcs

(f(x,y) =0 for every maximum flow from s to t if arc(x,y) is forbidden)



network N with capacities

F* = forbidden arcs

(f(x,y) =0 for every maximum flow from s to t if arc(x,y) is forbidden)

The red arc is in F*

In G=(B,W,E) connected where all vertices are free,
B and W are the only maximum weight stable sets
if and only if $F^* = \emptyset$
(no forbidden arcs in associated N)

In G=(B,W,E) connected where all vertices are free,
B and W are the only maximum weight stable sets
if and only if F*= Ø
(no forbidden arcs in associated N)

Remove from G all edges associated with forbidden arcs of N.

It remains connected components inducing partition $V_1, V_2, \ldots, V_q$ of V.

In each V_i the only maximum weight stable sets are

$$V_i \cap B \text{ and } V_i \cap W$$

In G = (B,W,E) with only free vertices, S is a maximum weight stable set if and only if S is stable and for any i $(1 \leq i \leq q)$ $S \cap V_i = V_i \cap B$ or $V_i \cap W$

In each V_i the only maximum weight stable sets are

$$V_i \bigcap B \text{ and } V_i \bigcap W$$

In G = (B,W,E) with only free vertices, S is a maximum weight stable set if and only if S is stable and for any i $(1 \leq i \leq q)$ $S \bigcap V_i = V_i \bigcap B$ or $V_i \bigcap W$

NB: Since B and W are two disjoint maximum weight stable sets in G, every d-transversal T must satisfy $|T \cap W| \geq d$, $|T \cap B| \geq d$ and hence $|T| \geq 2d$.

Construct auxiliary graph G* = (V*,A*) where

$$V^* = \{V_1, \ldots, V_q\}$$

$$A^* = \{(V_i, V_j) \mid \exists uv \in E, u \in V_i \cap B, v \in V_j \cap W, i \neq j\}$$

NB: G* has no oriented circuit

One can define relation $\prec$ on V by: $u \prec v$ if

$u \in W, v \in B$ and either u,v are in the same V_j

or there is a directed path from V_i to V_j in G*

NB : If $u \prec v$ then for any maximum weight stable set S in G we have $S \bigcap \{u, v\} \neq \emptyset$

Construct bipartite graph ^G=(B,W,Ê)
with $\hat{E} = \{uv : u \in W, v \in B, u \prec v\}$

Each edge uv has cost c(uv) = c(u) + c(v)

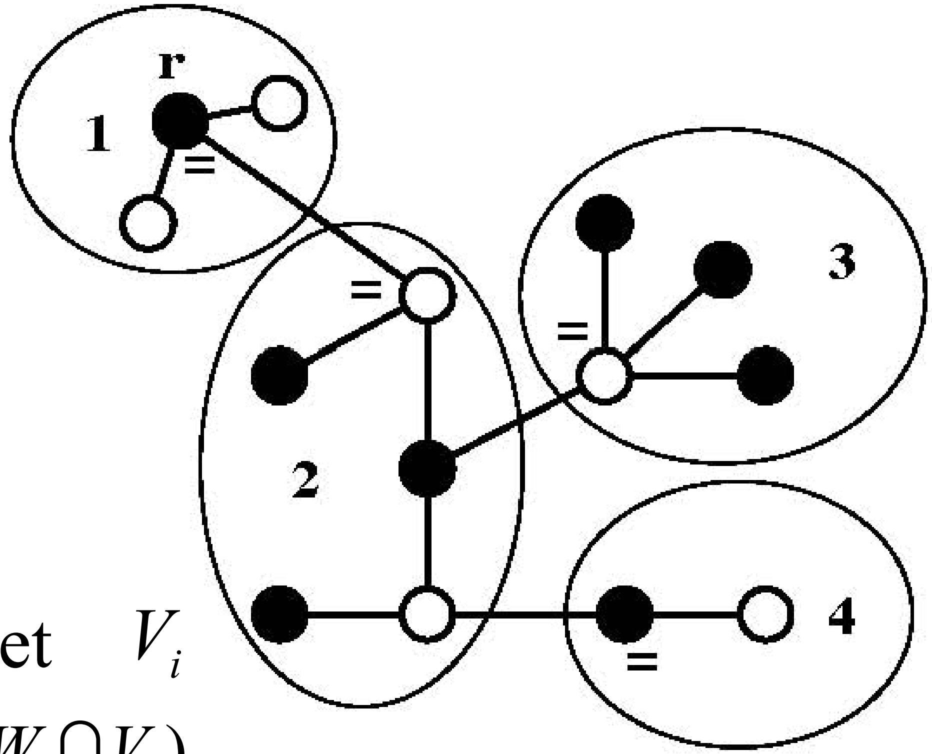
NB : If $u \prec v$ then for any maximum weight stable set S in G we have $S \cap \{u, v\} \neq \emptyset$

Construct bipartite graph ^G=(B,W,Ê)
with $\hat{E} = \{uv : u \in W, v \in B, u \prec v\}$

Each edge uv has cost c(uv) = c(u) + c(v)

T is a minimum cost d-transversal in G if and only if it consists of the endvertices of a minimum cost matching of size d in ^G.

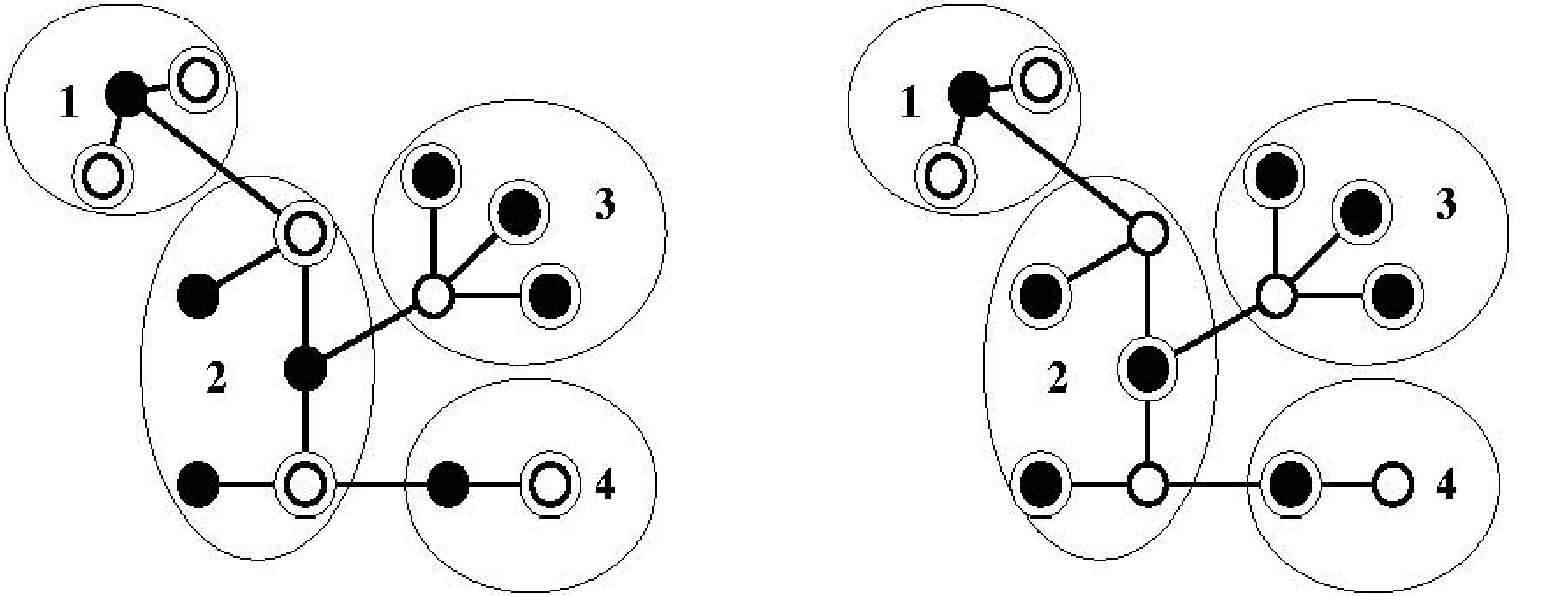
Weighted trees for illustration



In each subset V_i

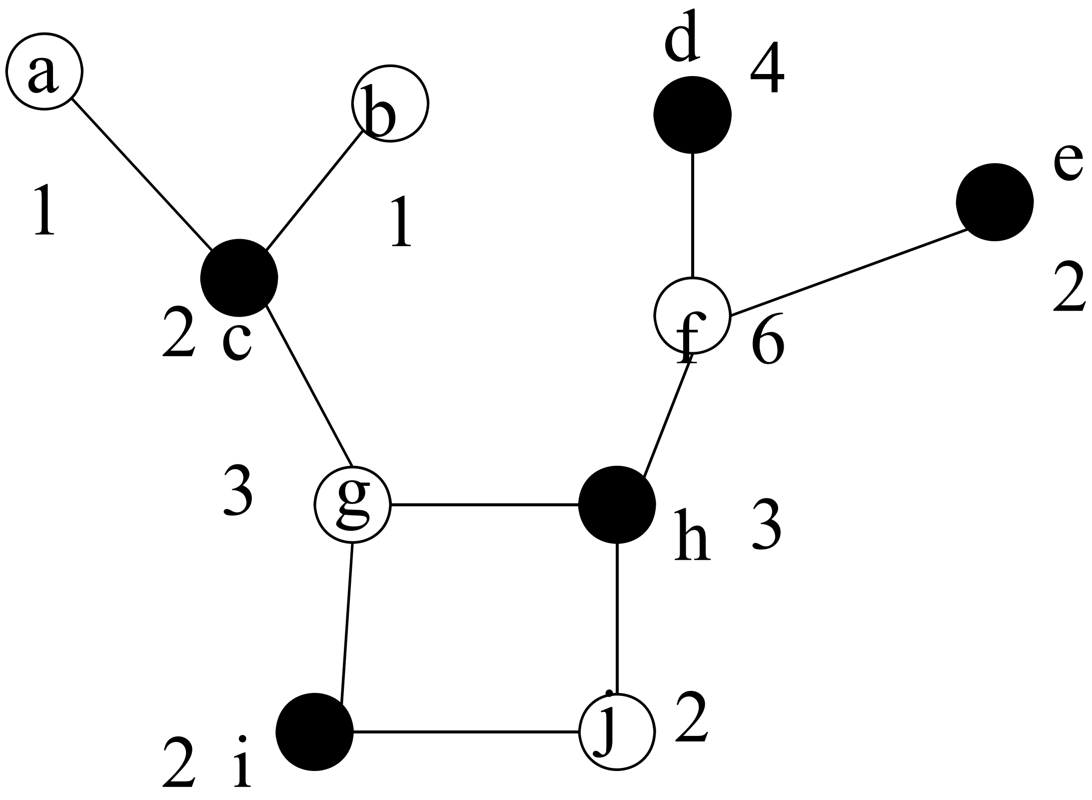
$$w(B \cap V_i) = w(W \cap V_i)$$

Weighted trees



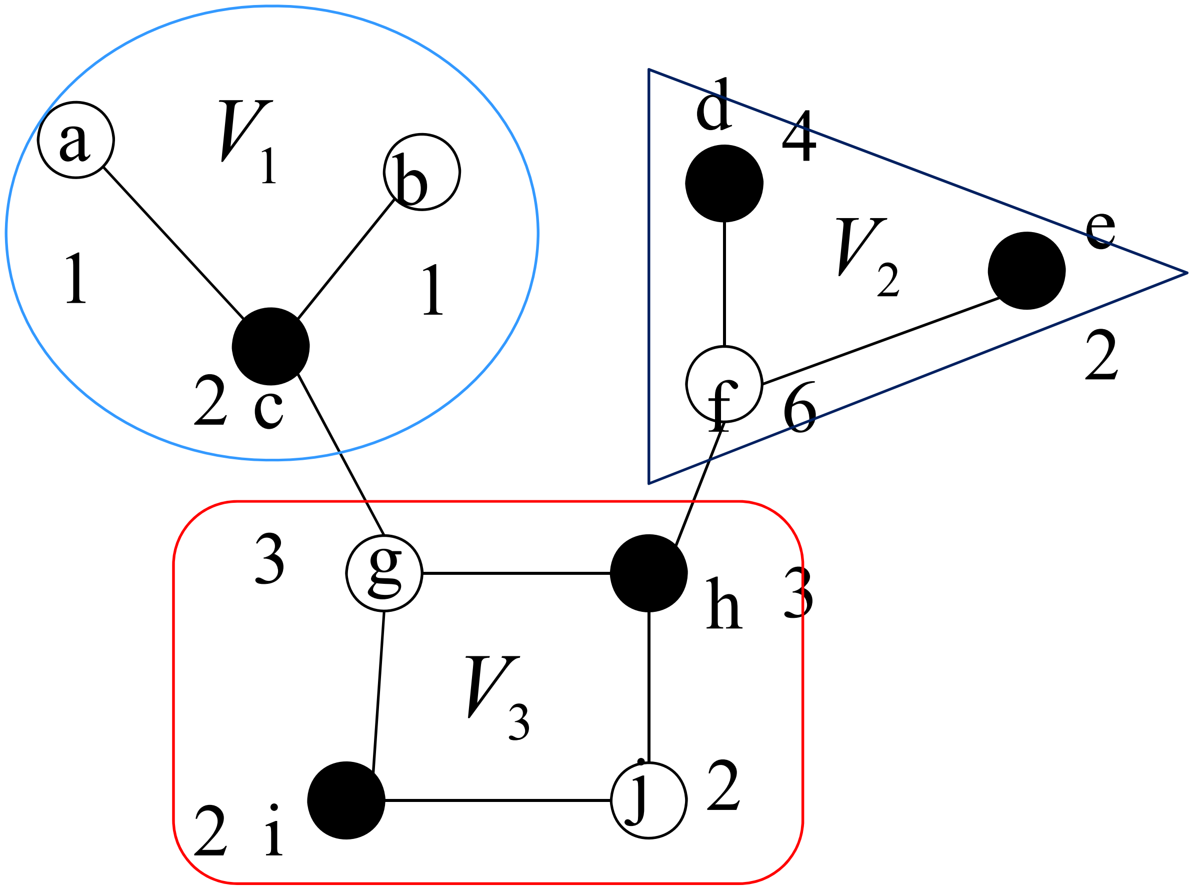
maximum-weight stable sets S

$$S \cap V_i = W \cap V_i \text{ or } B \cap V_i$$



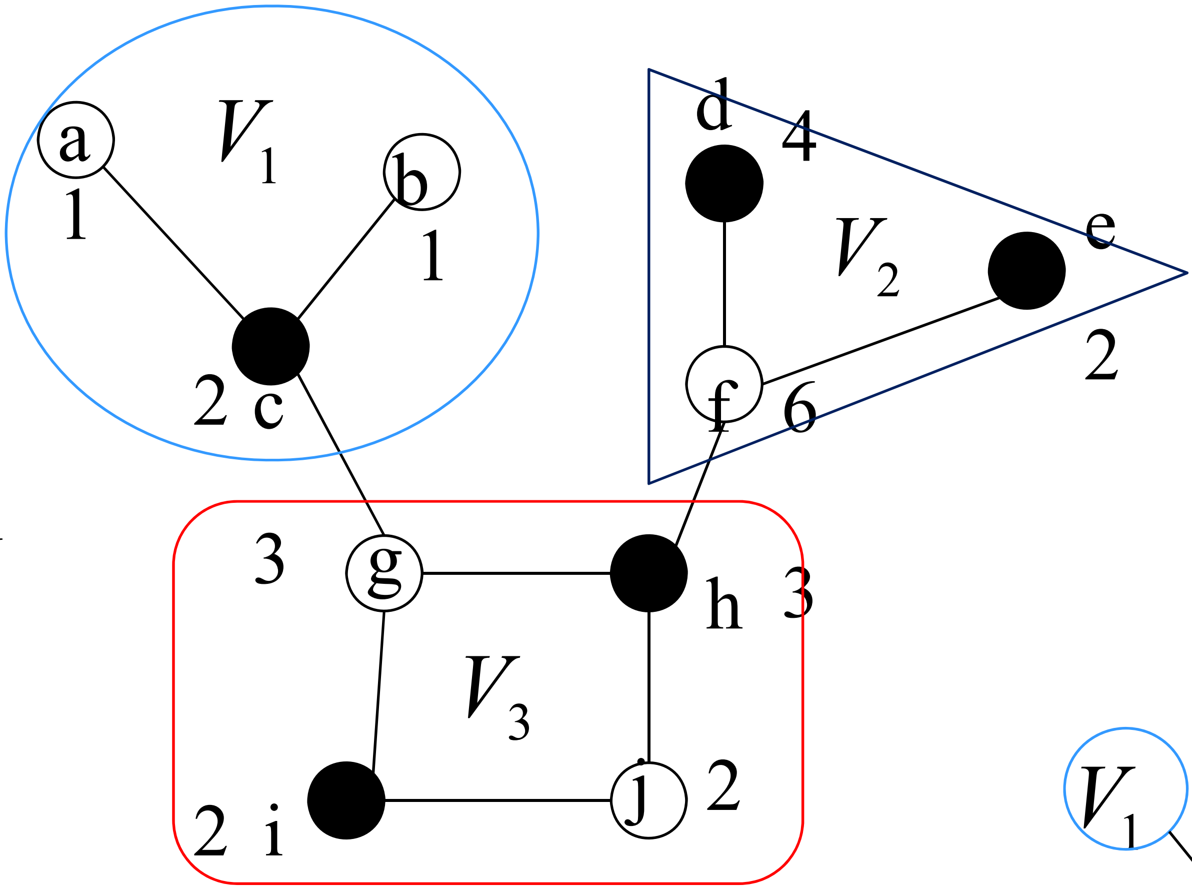
Weighted bipartite graph

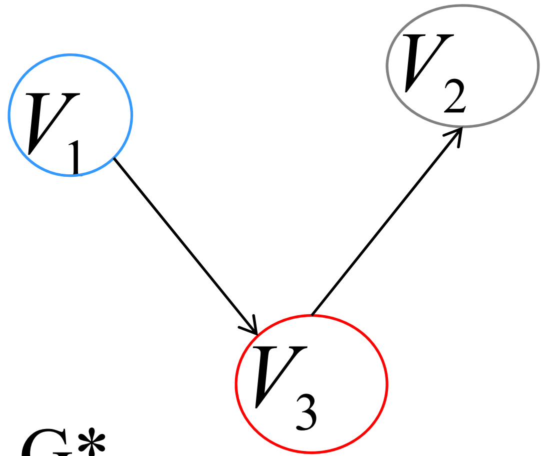
(all vertices are free)



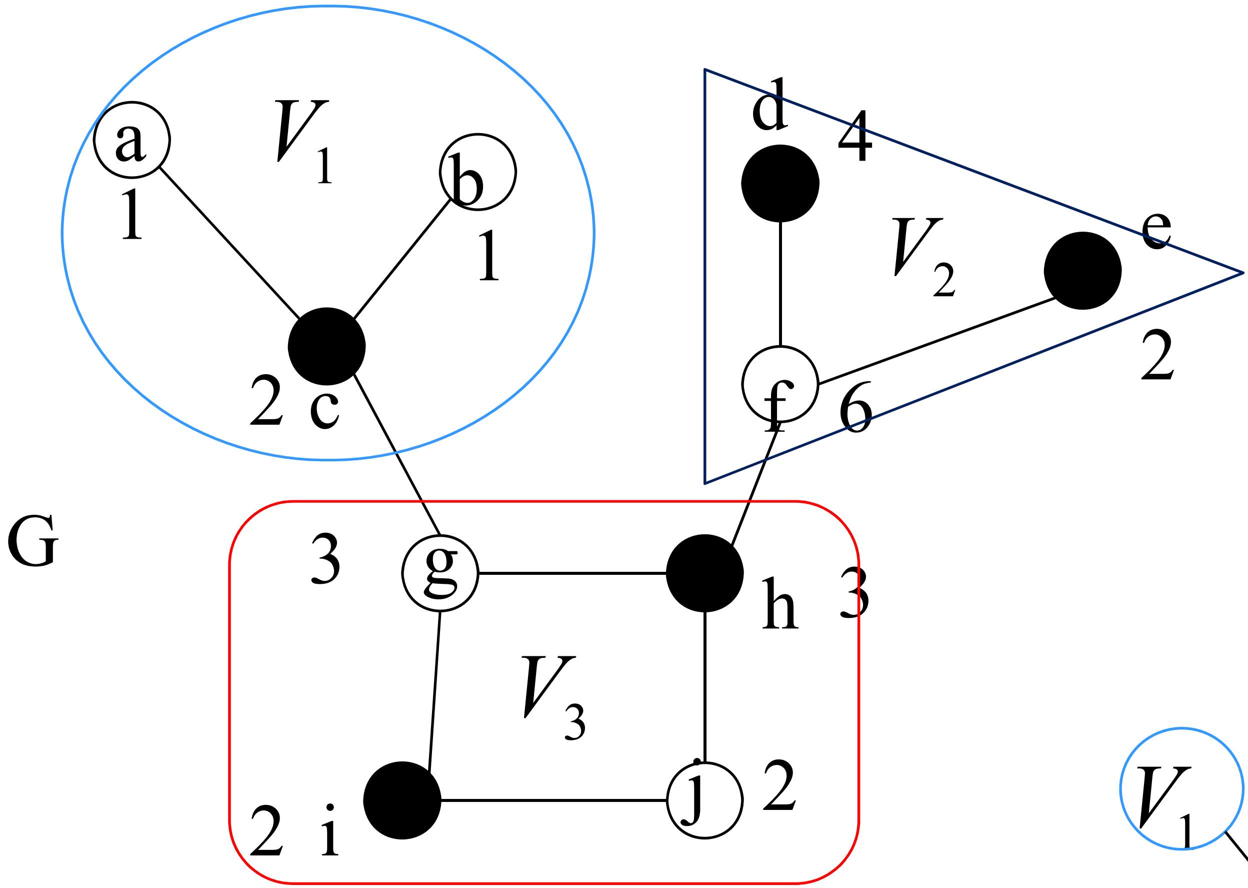
Partition into V_1, V_2, V_3

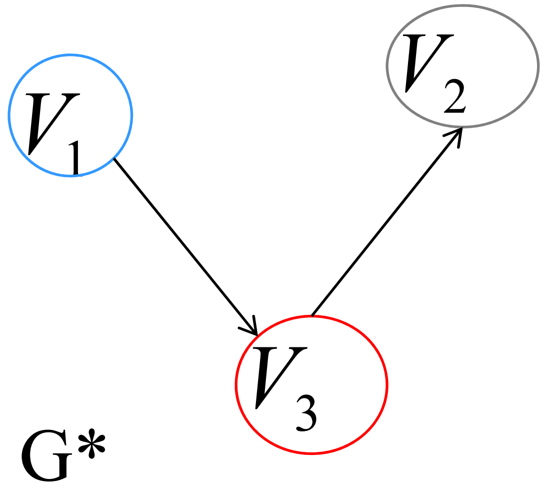
G





G*



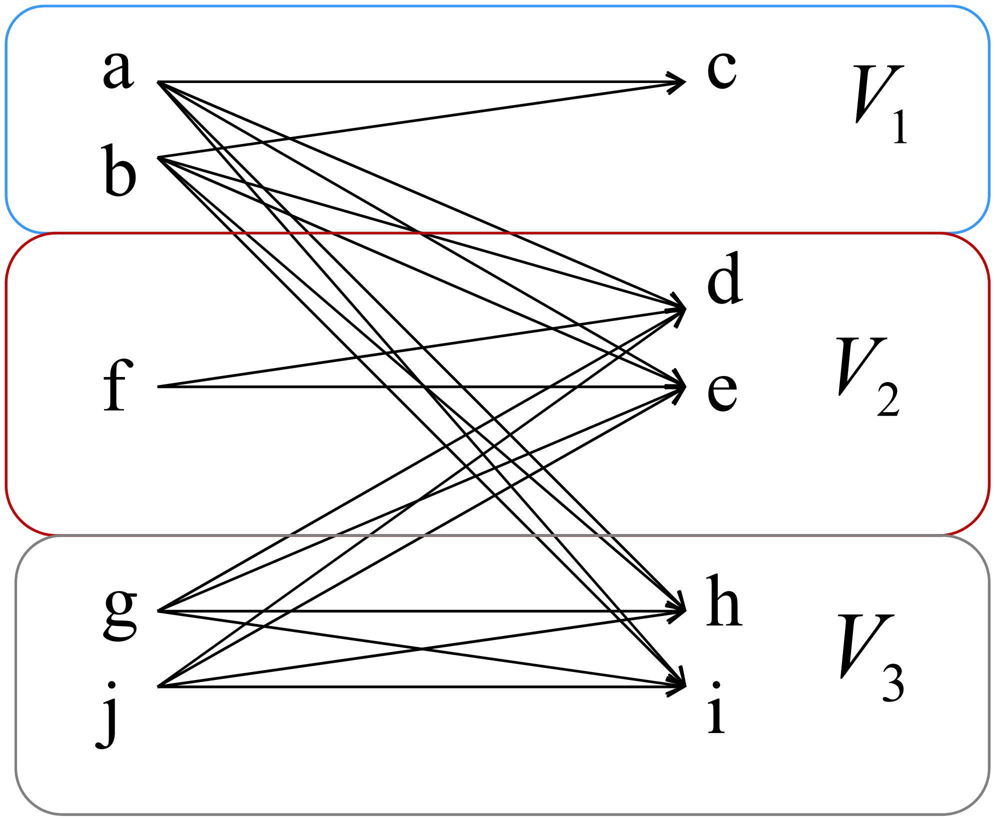


$$a, b \prec c, h, i, d, e$$

$$f \prec d, e$$

$$g, j \prec d, e, h, i$$

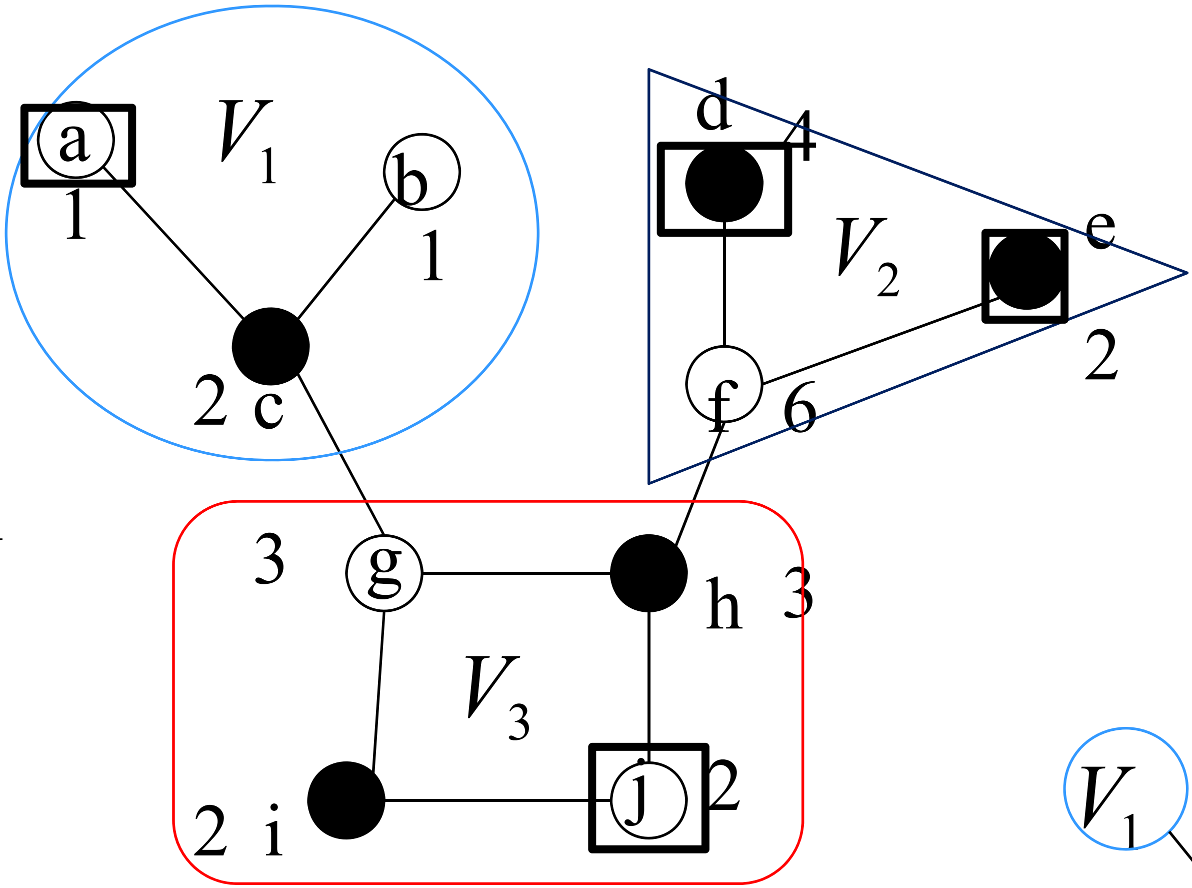
^G



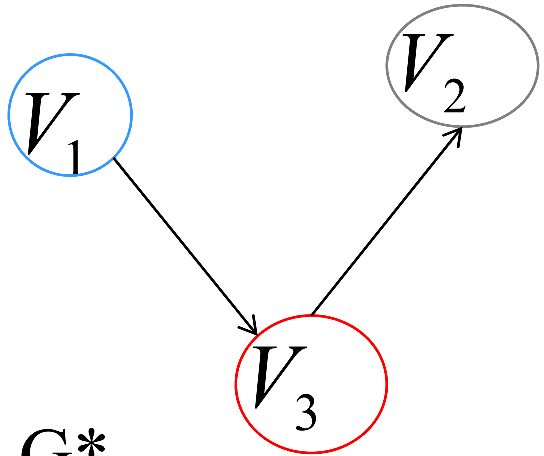
^G

Verification: ad,je matching with cost 1+4 +2+2 and size 2.

Is {a,d,j,e} a 2-transversal ?



G



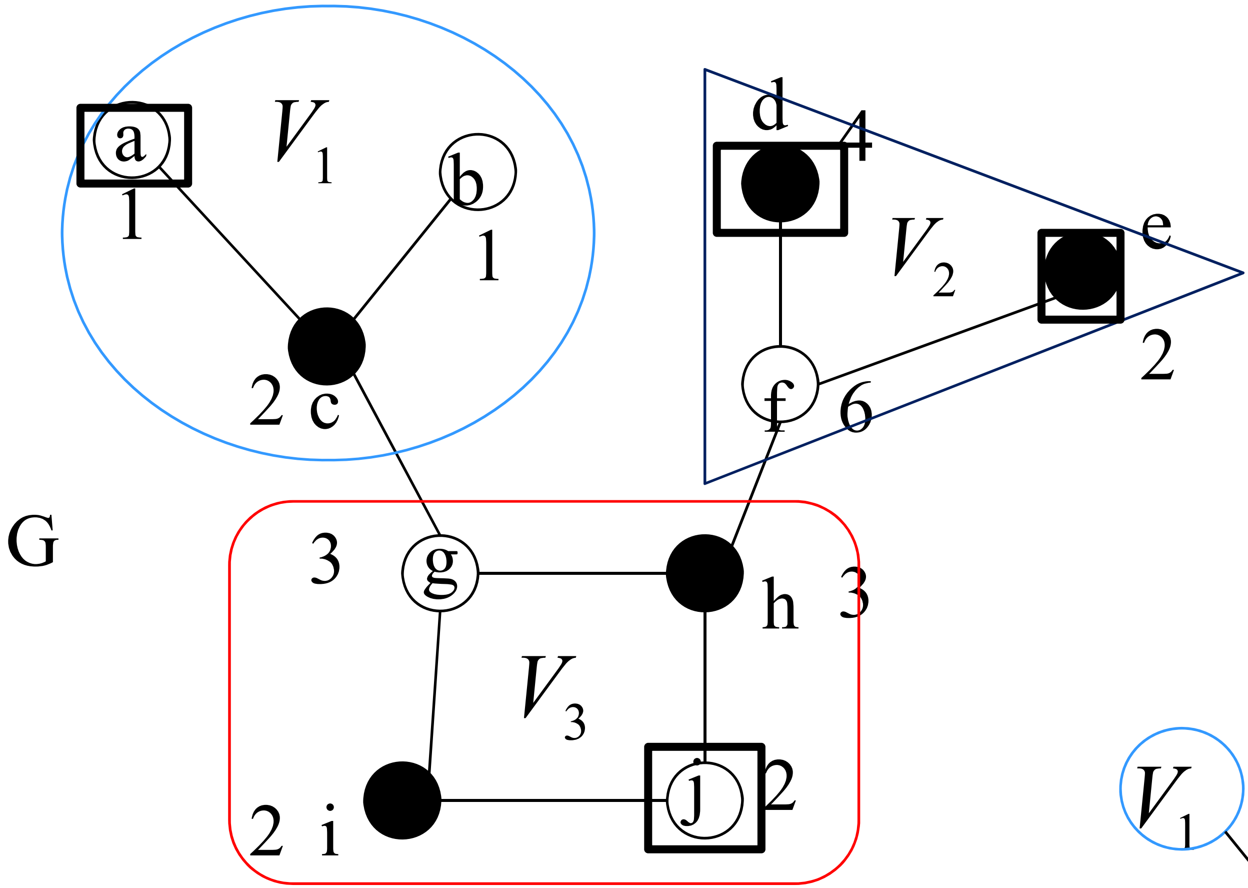
G*

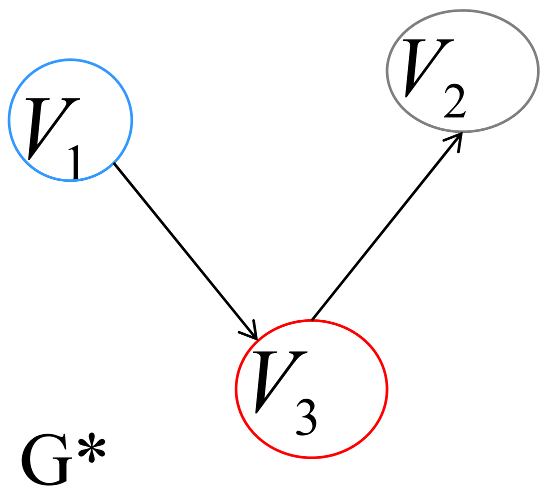
$$a, b \prec c, h, i, d, e$$

$$f \prec d, e$$

$$g, j \prec d, e, h, i$$

^G





If $a \notin S$ then $d \in S$

If $j \notin S$ then $e \in S$

¿ And if there are forced vertices in G ?

Introduce into ^G every forced vertex x
with weight w(x)
as an isolated edge xx'
with cost c(xx') =w(x).

¿ And if there are forced vertices in G ?

Introduce into ^G every forced vertex x
with weight w(x)
as an isolated edge xx'
with cost c(xx') =w(x).

Polynomial algorithm
(minimum cost bipartite matching)

The end…

…but much more is to be discovered.