Où l'on en verra de toutes les couleurs

… et avec des arguments de poids !

D. de Werra, EPFL

en collaboration avec : M. Demange, ESSEC

J. Monnot, LAMSADE

V. Paschos, LAMSADE

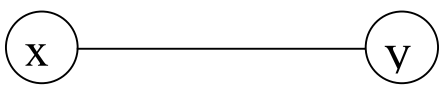
Ordonnancement chromatique

(chromatic scheduling)

modèles de coloration

pour problèmes d'ordonnancement

extensions pondérées

w(x)  w(y)

« haltère »

V = collection of jobs J_j

with processing times $w(J_j)$ (weights)

E = pairwise incompatibilities

(e.g.: non-simultaneity

or inclusion in different batches)

batch = collection S of compatible jobs

$w(S) = f(w(J_j) : J_j$ _ $S)$

= total completion time of jobs in batch S

Problem: Find a partition C of jobs of V

into batches $S_1,\ldots,S_k$ and a schedule

such that the total completion time

$C_{max}(\mathsf{C}) = g(w(S_1),\ldots,w(S_k)) = \min$!

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

job J_j	—	node J_j
J_r, J_s incompatible	—	edge $[J_r, J_s]$
batch	—	stable set
partition $\mathcal{C}$ into k batches?	—	node k – coloring
processing time $w(J_j)$	—	weight $w(J_j)$

"weighted coloring"

$$C_{max}(\mathcal{C}) \equiv \hat{K}(\mathcal{C}) = \text{weight or cost of coloring } \mathcal{C}$$

Example 1:

compatible jobs = jobs which may be processed on same machine

batch S_i = jobs assigned to machine i

$w(S_i) = \sum (w(J_j) : J_j \_ S_i)$

sequential processing of jobs of each batch

partition C into batches $S_1, \ldots S_k$

$C_{max}(\mathsf{C}) = \max \{w(S_1), \ldots, w(S_k)\}$

parallel processing of batches

Problem: Find a partition $\mathsf{C} = (S_1, \ldots, S_k)$

of jobs of V into batches S_i

(each S_i is a compatible set)

with $C_{max}(\mathsf{C})$ minimum

NB: k is fixed in this example!

(else take $k = |V|$ and $|S_i| = 1 \quad \forall\, i$)

$k \geq \chi(G)$ = chromatic number of G

Special case: $w(J_j) = 1 \quad \forall$ node J_j

$w(S_i) = \left(w(J_j) : J_j \in S_i\right) = |S_i|$

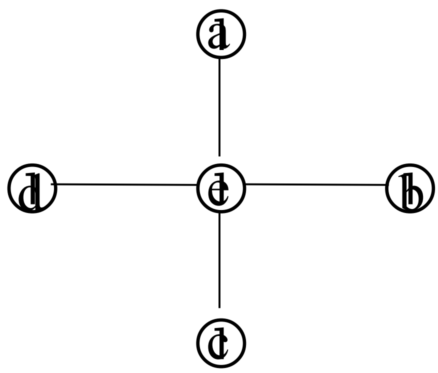
$C_{max}(C) = \max \{|S_1|, \ldots, |S_k|\}$

Problem: For k fixed

find a k-coloring $C = (S_1, \ldots S_k)$

such that

$\hat{K}(C) = \max \{|S_1|, \ldots, |S_k|\}$ is min



$k = 2 \quad S_1 = \{a,b,c,d\},\ S_2 = \{e\} \quad \hat{K}(C) = |S_1| = 4$

$k = 3 \quad S_1 = \{a,b\},\ S_2 = \{c,d\},\ S_3 = \{e\} \quad \hat{K}(C) = |S_1| = 2$

(Bodlaender, Jansen, Woeginger, 1994)

Example 2:

compatible jobs = jobs which may be in same batch

$w(S_i) = \max \{ w(J_j) \mid J_j \in S_i \}$

parallel processing of jobs in same batch

partition $\mathcal{C}$ into batches $S_1, \ldots S_k$

$C_{max}(\mathcal{C}) = = \quad (w(S_i) : i = 1,...,k)$

sequential processing of the batches

Problem: Find an integer k and

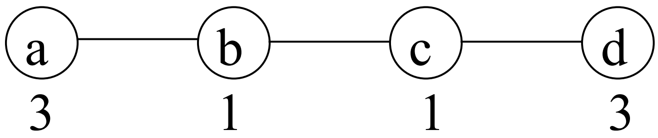
a partition $\mathcal{C} = (S_1, \ldots, S_k)$ of jobs

of V into k batches S_i (each S_i is

a compatible set) with

$C_{max}(\mathcal{C})$ minimum

NB: k has to be found ! $k \geq \chi(G)$



$\chi(G) = 2$

$S_1 = \{a,d\}, S_2 = \{b\}, S_3 = \{c\}$ $\quad k = 3 > \chi(G)$

$C_{max}(\mathcal{C}) = 3 + 1 + 1 = \min$!

Special case: $w(J_j) = 1 \quad \forall \quad J_j$

$w(S_i) = \max \{w(J_j) : J_j \in S_i\} = 1$

$C_{max}(\mathcal{C} = (S_1, \ldots, S_k)) = \quad (w(S_i) : i = 1, \ldots, k) = k$

Problem: Find a k-coloring of G

with k minimum

complexity and approximability of

weighted case: see

(Demange, de Werra, Monnot, Paschos, 2001

Time Slot Scheduling of compatible Jobs)

A "classical" application: satellite telecommunication

decomposition of traffic matrix

$T = (t_{ij})$ into permutation matrices $P^1,\ldots,P^n$

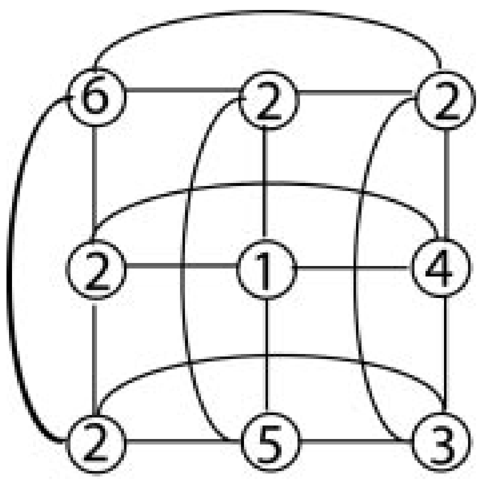
"switching modes"

such that $\left\{ \max_{i,j} \; p_{ij}^{s} \;\middle|\; s=1,\ldots,n \right\} = \min!$

$$\begin{array}{|c|c|c|}\hline 6 & 2 & 2 \\ \hline 2 & 1 & 4 \\ \hline 2 & 5 & 3 \\ \hline \end{array} = \begin{array}{|c|c|c|}\hline 6 & & \\ \hline & & 4 \\ \hline & 5 & \\ \hline \end{array} + \begin{array}{|c|c|c|}\hline & 2 & \\ \hline 2 & & \\ \hline & & 3 \\ \hline \end{array} + \begin{array}{|c|c|c|}\hline & & 2 \\ \hline & 1 & \\ \hline 2 & & \\ \hline \end{array}$$

$C_{max}(C) = \quad 6 \quad + \quad 3 \quad + \quad 2 \quad = 11$

Here : :

G = 

NB: NP-complete (F. Rendl, 1985)

Generalization of previous model

stable set (compatible) S in G

- $S' \subseteq S$ is also stable
- subgraph G(S) of G generated by nodes of S = isolated nodes

  = collection of node disjoint cliques of size 1

Idea: replace S by $S' \subseteq V$ such that subgraph $G'(S')$ of G generated by S'

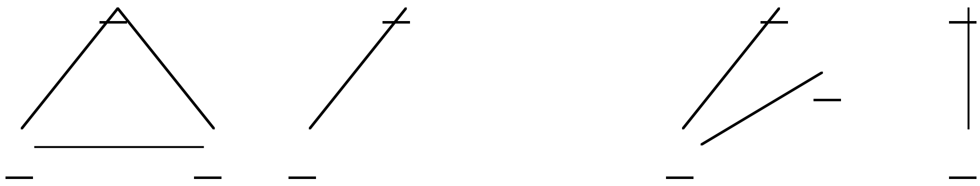
= collection of node disjoint cliques

Extension of basic model

def: In $G = (V,E)$, set $S \subseteq V$ **hypostable**

if S induces a collection of disjoint

cliques (without links)



hypostable   not hypostable

Hypochromatic number $\chi_h(G)$ = min k such that $\exists$ partition of V into k hypostable sets

NB: Determine whether $\chi_h(G) \leq 2$: NP-complete

Also called “subcoloring”

easy for complements of planar graphs

$(\chi_h(G) \leq 2)$ (Broersma, Fomin, Nesetril, Woeginger, 2002)

Such extensions of colorings
have been studied (generally unweighted)

M.O. Albertson, R.E. Jamison, S.T. Hedetniemi, S.C. Locke (1989)

J.L. Brown, D.G. Corneil (1987)

J. Fiali, K. Jansen, V.B. Le, E. Seidel (2001)

R. Dillon (1998)

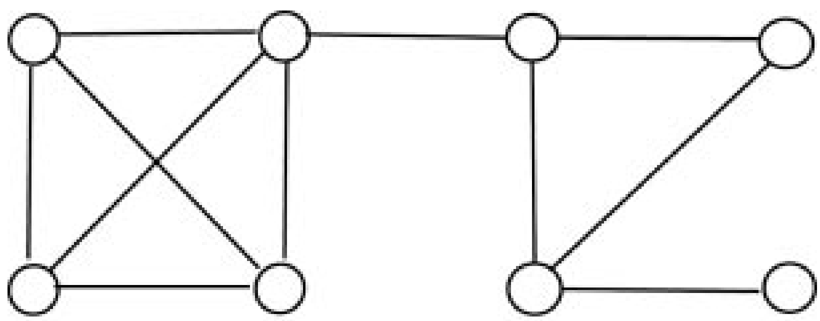
• • •

Solvable cases :

cactus: connected graph where any two cycles have ≤ 1 common node

If G = L(H) (line graph of cactus) then $\chi_h(G) \leq 3$

Block graph: every 2-connected component is a clique



If G = block graph, then $\chi_h(G) \leq 2$

Weighted case: weight w(v) ∀ v in G

clique K $w(K) = \left(w(v) \middle| v \lfloor\ K\right)$

S   hypostable set $w(S) = \max\left\{w(K) \middle| K \lfloor\ S\right\}$

$\mathsf{C} = (S_1,\ldots,S_k)$ $C_{max}(\mathsf{C}) = \left(w(S_i) \middle| i = 1,\ldots,k\right)$

hypocoloring

Interpretation:

J_i, J_j compatible	—	J_i, J_j can be processed simultaneously (assigned to different processors)
clique K	—	collection of jobs to be processed consecutively (on same processor) $w(K) = \left(w(v) \mid v \in K\right)$
hypostable set S	—	set of jobs (or of collections of incompatible jobs) $w(S) = \max\{w(K) \mid K \in S\}$

Problem: Find partition C of set V of jobs

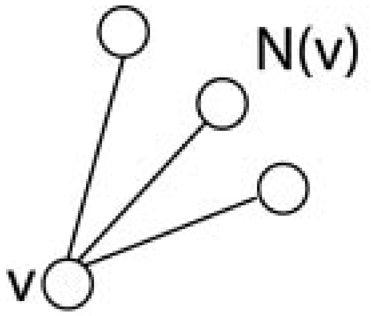
into batches (hypostable sets):

$$C_{max}(\mathcal{C}) = \min !$$

Property: In weighted graph G

$\exists$ k-hypocoloring S with min cost $\hat{K}(S)$

which has $k \leq \Delta(G) + 1$ colors



$\exists \; s \leq \Delta + 1$
missing in N(v)

color $l > \Delta + 1$

$$S_s' = S_s \cup \{v\} \qquad S_l' = S_l - v$$

NB: $w(S_1) \geq \ldots \geq w(S_s)$

$$w(v) \leq w(S_l) \leq w(S_s)$$

$$\uparrow \qquad \uparrow$$

$$v \in S_l \qquad s < l$$

$\Rightarrow$ $w(S_l') \leq w(S_l)$ $\qquad w(S_s') = w(S_s)$

no increase of cost

Repeat until $S' = (S_1', \ldots, S_k')$

with $k \leq \Delta + 1$

Brooks theorem: $\chi(G) \leq h$ if

G has $\Delta(G) = h$ and $G \neq$ clique (or odd cycle $h = 2$)

Improvement:

$\exists$ k-hypocoloring $\mathcal{S}$ with min cost $\hat{K}(\mathcal{S})$

and $k \leq \Delta(G)$

Sketch of proof: $\mathcal{S} = (S_1,\ldots,S_k)$ opt k-hypocoloring with $k \leq \Delta(G) + 1$ and $|S_k|$ minimum.

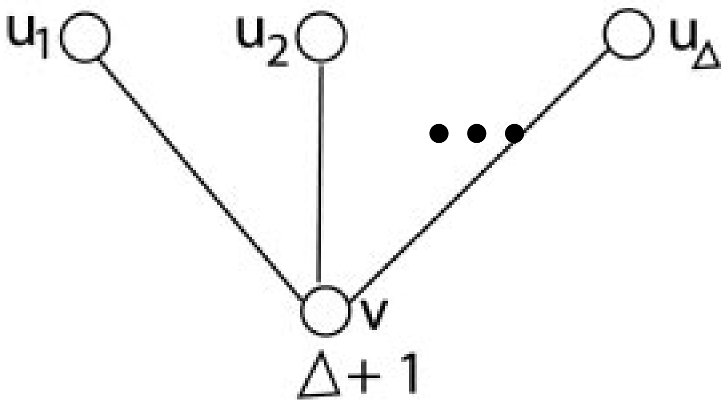
If $k \leq \Delta(G)$: OK

$k = \Delta(G) + 1$ let $v \llcorner S_k$

If $\exists$ color $s \leq \Delta(G)$ missing in $N(v)$

recolor v with $s \Rightarrow$ better coloring. Impossible

Hence colors $1, 2, \ldots, \Delta(G)$ occur in $N(v)$



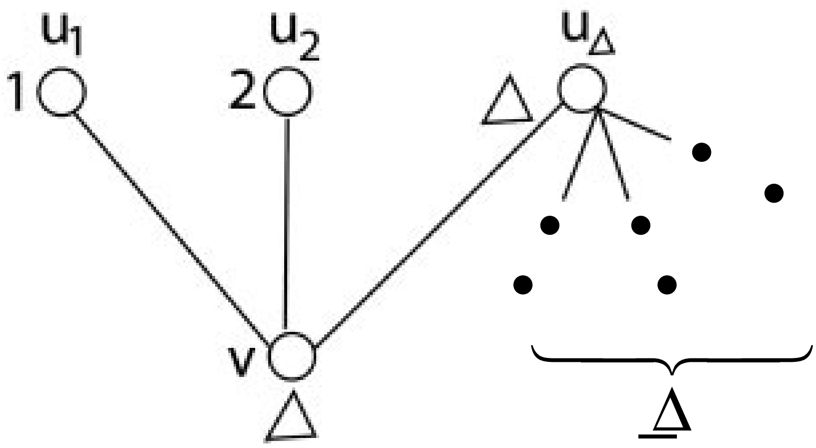
$\exists$ color $s \leq \Delta$ missing in $N(u_\Delta)$

If $s < \Delta$ recolor u_Δ with s

and v with Δ $\Rightarrow$ Better coloring. Impossible

Hence $s = \Delta$ missing in $N(u_\Delta)$

Recolor v with Δ !



Repeat for all nodes in $S_k \leftarrow \Delta + 1$

$\rightarrow \Delta$-coloring $S' = (S'_1, \ldots, S'_k)$

$$w(S'_\Delta) \leq w(S_\Delta) + w(S_{\Delta+1})$$

Better coloring. Impossible

Bound Δ best possible:

$\forall \; p > 0 \;\; \exists$ tree G with $\Delta(G) = p$

and with optimum k-hypocoloring

with $k = p$ colors

Complexity of weighted hypocoloring

NP-complete for graphs G with

$\Delta(G) = 3$ and $w(v) \in \{a, b\}$

$\exists$ polynomial algorithm for
trees with bounded degree

"special case": graphs with $\Delta(G) = 2$

A special case: $\Delta(G) = 2$

G = cycles and chains

$w(v) \geq 0 \quad \forall$ node v

Proposition: If G = collection of chains, then

$\exists \ G'$ = single cycle such that $\forall \ r$

G' has 2-hypocoloring $\mathcal{C}'$ with $\widehat{K}(\mathcal{C}') \leq r$

iff G has 2-hypocoloring $\mathcal{C}$ with $\widehat{K}(\mathcal{C}') \leq r$.

2 —— 1 —— 3 —— 4 - - - 0 - - - 0 - - - 5 —— 1 —— 6

Consequence: We may assume G = disjoint cycles

NB: S hypostable = nodes, edges, triangles

for $e = [x,y] \quad w(e) = w(x) + w(y)$

$\exists$ optimal 2-hypocoloring

$w(S_1) \geq w(S_2) \quad S_2$ contains no triangle

Basic idea: for fixed $p \geq q$ use algorithm

$A(p, q)$ which determines if $\exists \ \mathcal{C} = (S_1, S_2)$

with $w(S_1) = p, \quad w(S_2) = q$

Properties used in Δ (p,q):

A) If $w(v) > q$, then $v \in S_1$

B) If x, y, z consecutive on a P_3 (_—_—_)
with $x, y \in S_i$, then $z \in S_{3-i}$

C) If for $e = [x,y]$, $w(e) > p$, then x, y not both in S_1 ("color 1 forbidden for e")

D) If $w(e) > q$, then x, y not both in S_2 ("color 2 forbidden for e")

E) If $a_1, a_2, \ldots, a_s$ = chain with $a_1, a_s \in S_i$ (s odd)
or $a_1 \in S_i$, $a_s \in S_{3-i}$ (s even), then
∃ 2-hypocoloring such that colors alternate on chain

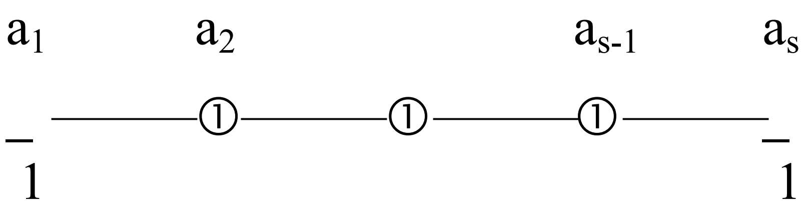
F) If $a_1, a_2, \ldots, a_s$ = chain with $a_1, a_s \in S_i$ (s even)
or $a_1 \in S_i$, $a_s \in S_{3-i}$ (s odd) then
∃ 2-hypocoloring such that $[a_1, a_2]$ gets a feasible color

Apply properties until a

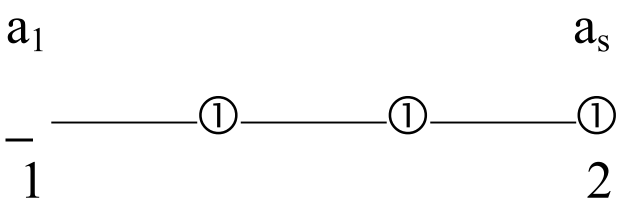
2-hypocoloring is obtained

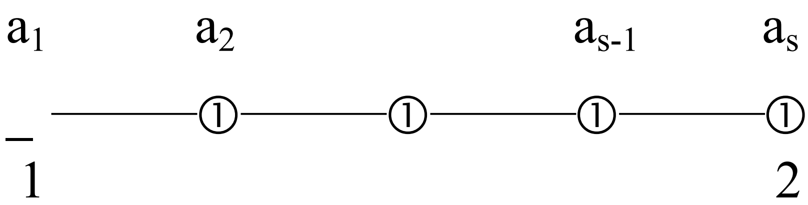
or a contradiction.

Record solution if best so far



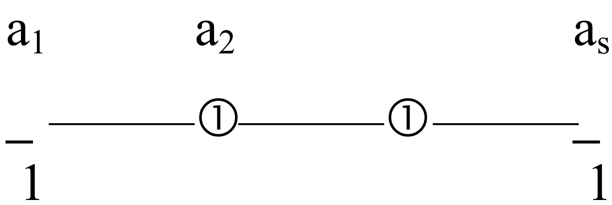
alternate colors 1 & 2





$[a_1, a_2] \rightarrow$ color 1

$w(a_1) + w(a_2) \leq p$ (else a_2 5 S_2)



Apply properties A) – F) until

a 2-hypocoloring is obtained

(or a contradiction).

Record solution if best so far

Property: max {w(v) : v 5 V) ″ w(S_1) ″

″ max { max {w(e) : e 5 E} , max{w(K_3) : K_3 5 G}}

where K_3 is a triangle in G

Algorithm: Start with smallest p

(and smallest $q \leq p$) and apply A(p, q)

to get smallest q for which C = (S_1, S_2) exists.

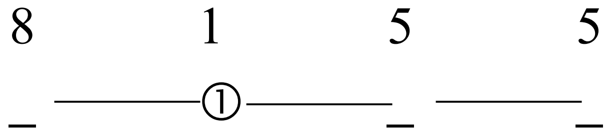
Increase p to next possible value and

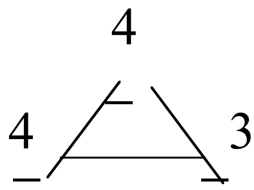
repeat A(p, q) with minimum q.

Stop when p is at maximum possible value.

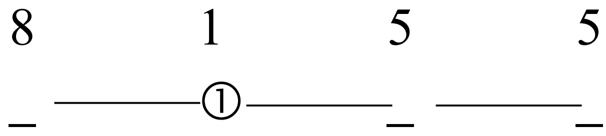
Complexity: $O(n^2)$

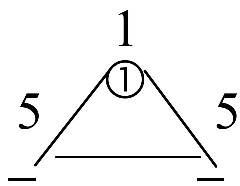
Examples:



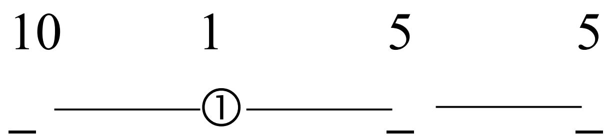


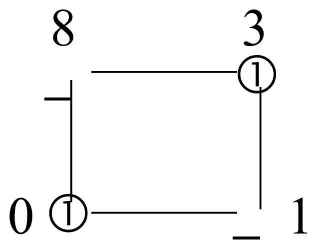
$w(S_1) = 11$ $w(S_2) = 1$





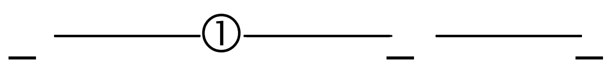
$w(S_1) = 10$ $w(S_2) = 1$

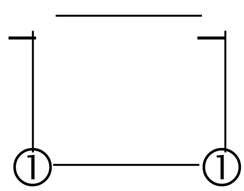




$w(S_1) = 10$
$w(S_2) = 1$
optimum

$w(S_1) = 8$
$w(S_2) = 3$
optimum





$w(S_1') = 11$

$w(S_2') = 1$

optimum

↑

not optimum

A special case:

2-restricted hypostable sets:

collection of cliques of cardinality ≤ 2

"nodes and edges"

Property: ∃ optimal k-hypocoloring

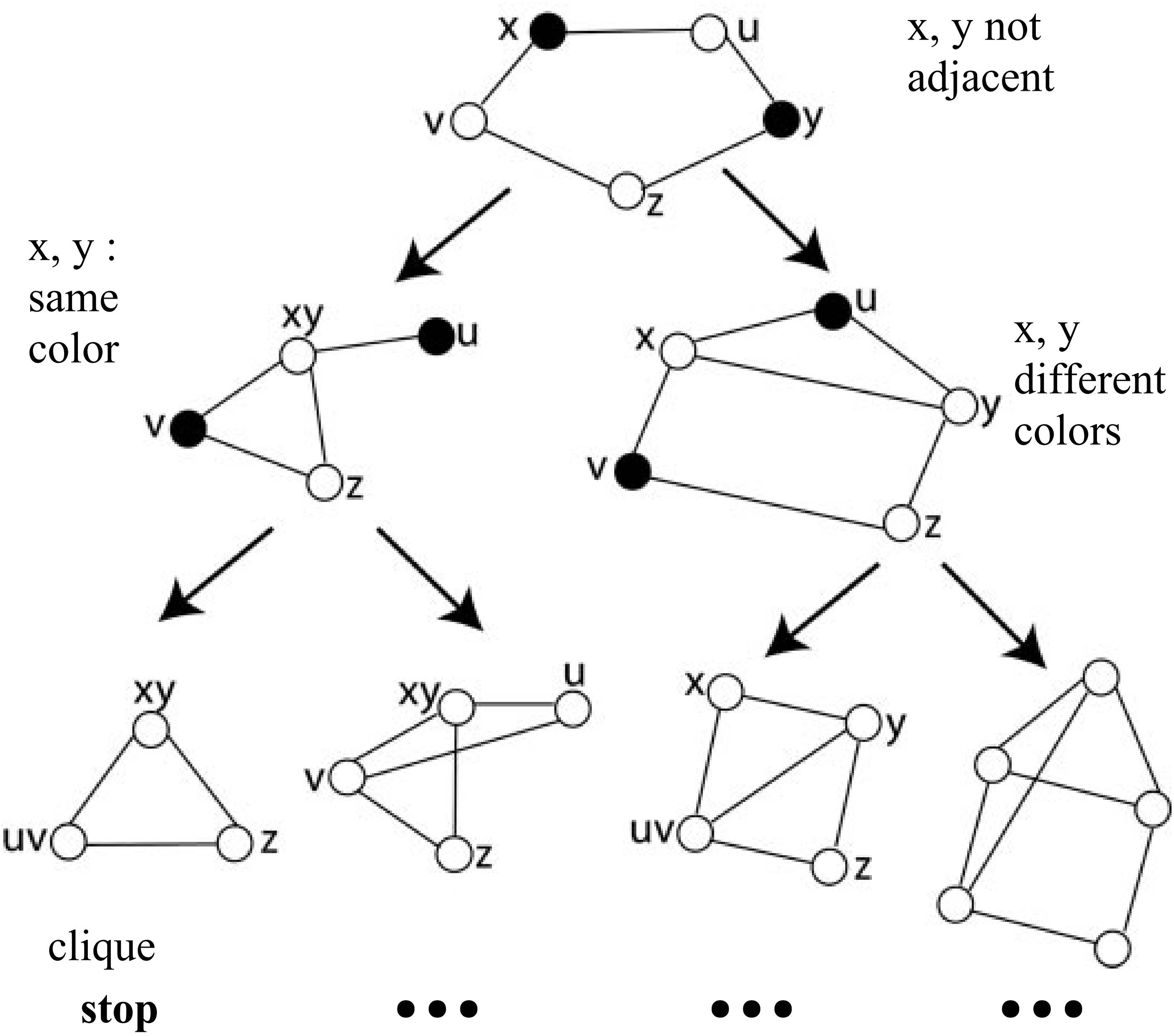
with k ≤ A(G)

For graphs without triangles

∃ enumeration algorithm COCA

(contract or connect algorithm)

"Light" version: usual colorings



G triangle-free: hypostable sets

"nodes and edges"

C

For G with **x, y not linked**: partition of colorings

a) x, y in same S_i: $G \leftarrow G_{X \equiv Y}$

x, y condensed into xy

$w(xy) = \max \{w(x), w(y)\}$

b) x, y not in same S_i: $G \leftarrow G + [x, y]$

H

For G with **edge [x, y]** : partition of hypocolorings

a) x, y in same S_i: $G \leftarrow G_{X \equiv Y}$

[x, y] condensed into xy

$w(xy) = w(x) + w(y)$

edges adjacent to x y are <u>**blocked**</u>

b) x, y not in same S_i:

$G \leftarrow G$ with [x, y] <u>**blocked**</u>

Initialization: G without triangles

weights $w(v)$; $L = \{G\}$: list of graphs
to examine

while $L \neq \emptyset$ choose G^* in L

If G^* has a free edge [x, y]

then apply separation H (introduce 2 modified G'_S into L and remove G^*)

else (all edges blocked)

if $G^* \neq$ clique, **then** apply separation C (introduce 2 modified G'_S into L and remove G^*)

else (G^* = clique with all edges blocked)

$w(G^*) = \left(w(v) \mid v \in V(G^*) \right)$

update best solution if necessary; remove G^*

COCA finds optimum (weighted)

hypocoloring in any graph G

if hypostable sets are defined

as "nodes and edges"

(node disjoint cliques of size ≤ 2)

Some extensions:

Hypostable set S: every connected component is a clique

$S' \subseteq S$ is also hypostable

(hypostability = hereditary property)

More generally: let P be hereditary property

S is a **P-constrained** set if every connected component C_s of S satisfies P.

Examples: P = "C_s is a clique"

P = "C_s is planar" (cf VLSI)

$C(S) = \{C_1,\ldots,C_r\}$

connected components of S

$V(C_s)$ = nodes of C_s

$f(C_s) = f(w(v) : v \; 5 \; C_s)$

$w(S) = \max \{f(C_s) : C_s \; 5 \; C(S)\}$

Hypothesis: for $C_s = \{v\}$, $f(C_s) = w(v)$

def: A **P-constrained k-coloring**

$\mathcal{C} = (S_1,\ldots,S_k)$ of $G = (V, E)$ is a partition of V into k P-constrained sets

def: cost of $\mathcal{C}$ $\hat{K}(\mathcal{C}) = \left(w(S_i) : i = 1,\ldots,k\right)$

with $w(S_i) = \max\{f(C_s) : C_s \in \mathcal{C}(S_i)\}$

Examples :

A) weighted hypocolorings

P	=	C_s is a clique
$f(C_s)$	=	$\left(w(v) : v \in V(C_s)\right)$
$w(S_i)$	=	$\max\ \{f(C_s) : C_s \in \mathcal{C}(S_i)\}$

B)

P	=	$\emptyset$ $w(v) = 1$ $\forall v \in V$
$f(C_s)$	=	$\lvert V(C_s) \rvert$
$w(S_i)$	=	largest # nodes in connected component of S_i
$\mathcal{C}$	=	partition of V into arbitrary $S_1,\ldots,S_k$

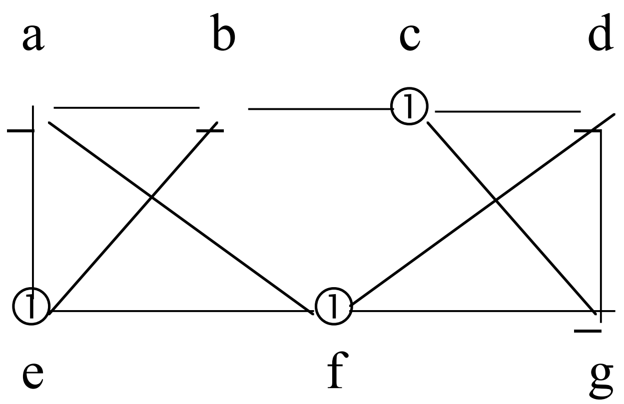
Remark: $\mathsf{C} = (S_1,\ldots,S_k)$

partition into arbitrary subsets

$$\chi(G) \leq \hat{K}(\mathsf{C}) = \ {}_{i=1}^{k} \max\left\{ |V(C_s)| : C_s \lfloor\ C(S_i) \right\}$$

In fact

$$\chi(G) = \min_{\substack{(S_1,\ldots,S_k) \\ \text{partition of } V(G)}} \ {}_{i=1}^{k} \max\left\{ |V(C_s)| : C_s \lfloor\ C(S_i) \right\}$$



$S_1 = a\ b\ d\ g$
$S_2 = c\ e\ f$

$C(S_1) = \{a\ b,\ d\ g\}$ $\qquad w(S_1) = 2$

$C(S_2) = \{\ c,\ e\ f\ \}$ $\qquad w(S_2) = 2$

$\chi(G) \leq 4$

Property: $G = (V, E)$ weighted graph

$\mathcal{C} = (S_1,\ldots,S_k)$ partition of V

into arbitrary $S_1,\ldots,S_k$

$w_{max}(U) = \max\{w(v) : v\ 5\ U\}$ – $U \subseteq V$

then

$$\min \hat{K}(\mathcal{C}) \leq \ {}^{k}_{i=1} w_{max}(S_i)\ \max\left\{\left| V(C_s) \right| : C_s \lfloor\ C(S_i)\right\}$$

$\mathcal{C}$ = P-constrained coloring

****

Alternate definition of weighted chromatic number:

(S_i = stable set; $w(S_i) = \max\ \{w(v) : v\ 5\ S_i\}$)

$$\min \hat{K}(\mathcal{C}) = \min \ {}^{k}_{i=1} w_{max}(S_i)\ \max\left\{\left| V(C_s) \right| : C_s \lfloor\ C(S_i)\right\}$$

$\mathcal{C}$: coloring $S_1,\ldots,S_k$
arbitrary
partition

¿ coloring algorithm for the unweighted case ?

A “special” case:

P-constrained chromatic number $\chi_p(G)$

$$= \min \widehat{K}(\mathcal{C}) = \sum_i \left(w(S_i) : S_i \in \mathcal{C}\right)$$

$\mathcal{C} = (S_1,\ldots,S_k)$ partition into P-constrained subsets

$w(v) = 1 \quad \forall\, v \in V \quad ; \quad f(C_s) = \max\,\{w(v) : v \in C_s\} = 1$

$w(S_i) = \max\,\{f(C_s) : C_s \in C(S_i)\,\} = 1$

Property: For $G = (V, E)$ weighted

with $w(v) > 0\ \forall\, v \qquad w(v) \in \{t_1, t_2,\ldots, t_r\}$

every optimal P-constrained coloring $S_1^*,\ldots,S_k^*$

with $f(C_s) = \max\,\{w(v) : v \in C_s\}$

satisfies

$$k \le 1 + r\,\left(\chi_p(G) - 1\right)$$

Sketch of proof:

Assume $w(S_1^*) \geq \ldots \geq w(S_k^*)$; let $q = \chi_p(G)$

To be shown $w(S_i^*) > w(S_{i+q-1}^*) \quad \forall\, i \leq k - q$

Take smallest i with $w(S_i^*) = \ldots = w(S_{i+q-1}^*) ?\, w(S_k^*)$.

Then $w(S_i^*) = \ldots = w(S_{i+q-1}^*) = t_s = \max\ \{w(v) : v\ 5\ G'\}$

where G' = subgraph generated by $S_i^* \cup \ldots \cup S_k^*$.

But $\chi_p(G') \leq \chi_p(G) = q$, so $\exists$ P-constrained

coloring $S_i', \ldots, S_{i+q-1}'$ of G' with $i + q - 1 < k$.

Assume $w(S_i') ? \ldots ?\ w(S_k')$; then $w(S_i') = w(S_i^*)$

and $w(S_{i+s}') \leq w(S_{i+s}^*)$ for s = 1,…,q-1.

Setting $S_j' = S_j^*$ for j = 1,…, i-1,

we get a P-constrained coloring C'

with $i + q - 1 < k$ colors; since $w(S_k^*) > 0 = w(S_k' = \emptyset)$,

we have $\hat{K}(C') < K(C^*)$.

Contradiction !