

OR 2

13-APR-2017

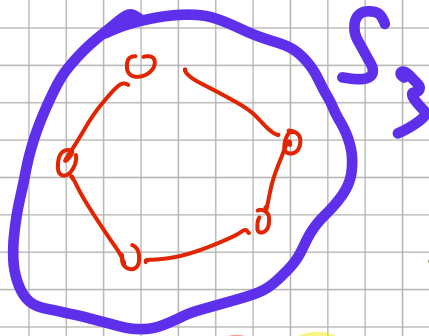
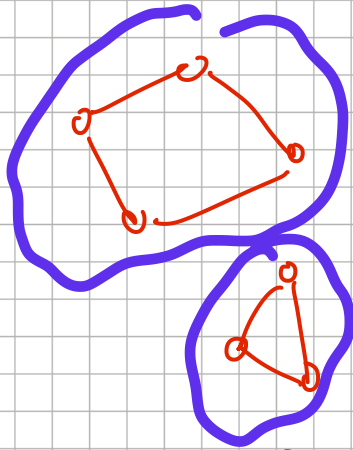
$$\min \sum_{e \in E} c_e x_e$$

$$\sum_{e \in \delta(v)} 1 \cdot x_e = 2, \quad \forall v \in V$$

$$x_e \in \{0, 1\}, \quad \forall e \in E$$

$S_1$

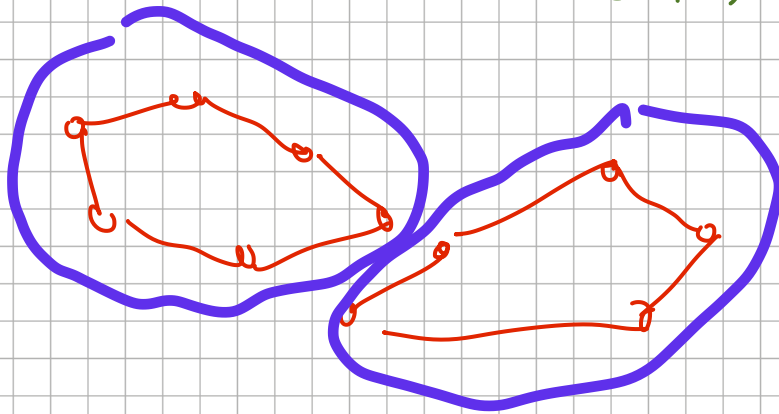
ITER. #1



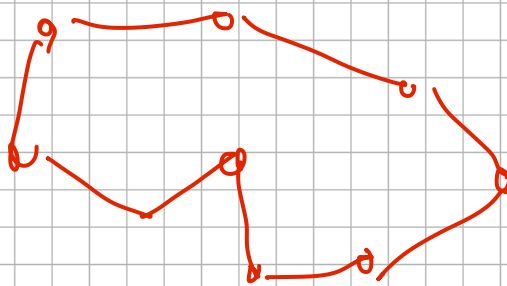
sec's

$$\sum_{e \in E(S_k)} 1 \cdot x_e \leq |S_k| - 1$$

ITER #2



...



only one conn. comp:  
STOP!

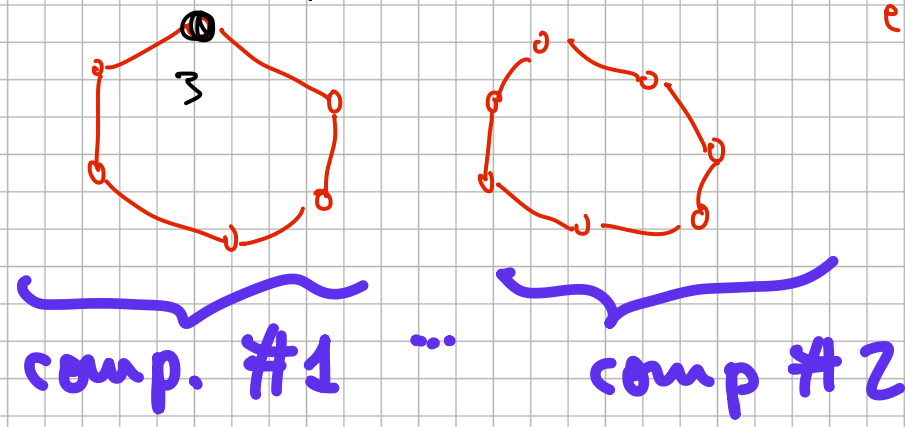
BENDER'S "loop" method

• How to detect the connected comp.?

$x_{star} \leftarrow CPX.getx(\dots)$

$comp[3] = 1$

$x_e^* = 1$



$comp[j] = \# \text{ comp. containing vertex } j, \# j$

$n. \text{ components} \rightarrow ncomp$

Adding  $S \in C$  to  $CPLEX$ 's model

if  $ncomp \leq 1$  then "STOP"

for  $k = 1$  to  $ncomp$  do

$num = 0; rhs = -1;$

for  $(i \in V : comp[i] = k)$  do

$rhs++;$

for  $(j \in V : j > i \ \& \ comp[j] = k)$  do

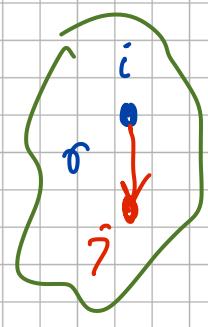
$index[num] = xpos(i, j);$

$coeff[num] = 1.0;$

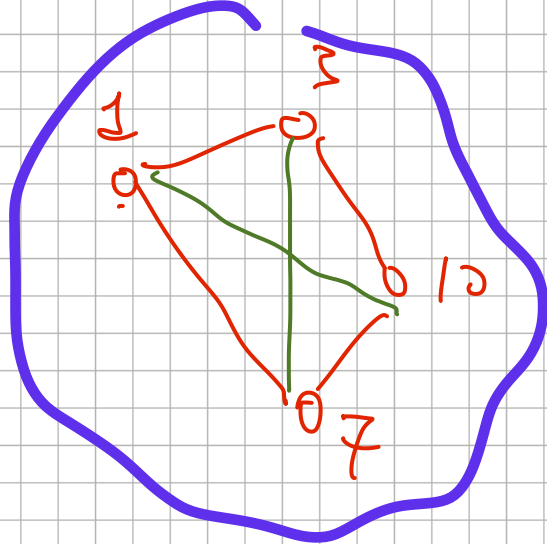
$num++$

for  $CPX.addrows(\dots, num, index, coeff, rhs)$

$k$



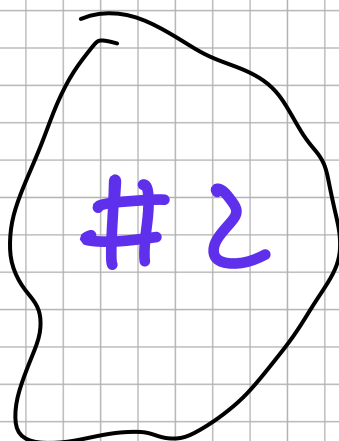
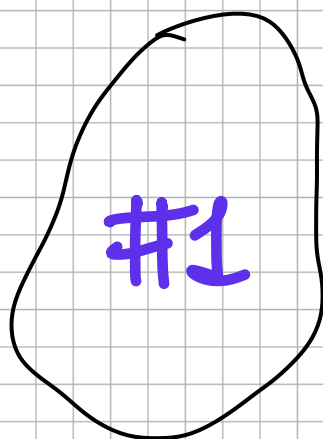
$k$



$$x_{13} + x_{3,10} + x_{10,7} + x_{7,1} \leq 3$$

"weak sec"

$$+ x_{37} + x_{1,10}$$



only 2 comp. s

⇒ forget about one of the two !!

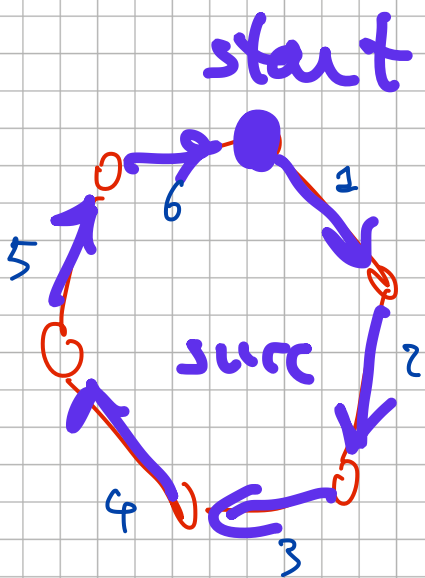
$n_{cols} = \text{CPX getnum cols}(\dots)$

⊗ int \* index = .. calloc( $n_{cols}$ ..)  
 double \* coeff = .. " "

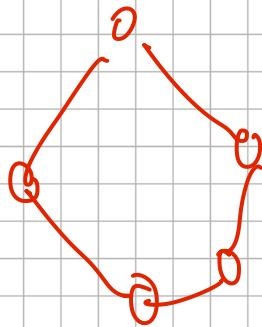
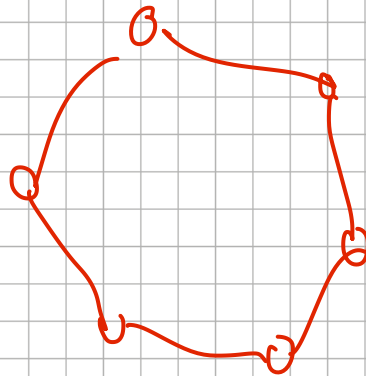
• How to compute the conn. components?

⇒  $O(n^2)$  very simple alg  
⇒ KRUSKAL implement.  
for SST

⇒ specialized alg.



~~X~~



⇒  $comp[i] = -1 \quad \forall i$

⇒ succ[i]