THE MIN-COST FLOW PROBLEM

Esteve Codina

Universitat Politècnica de Catalunya

Grau d'Enginyeria Informàtica FIB

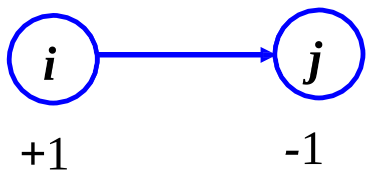
CONTENTS

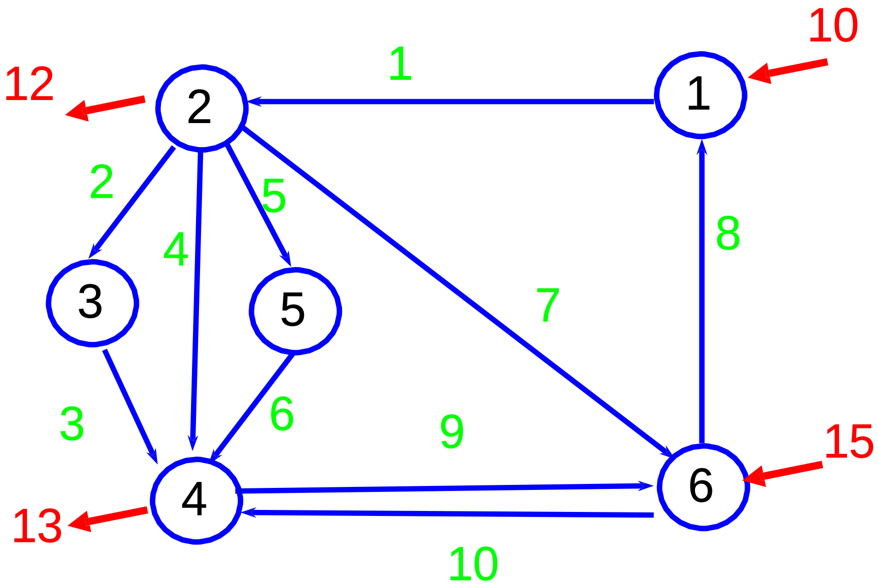
UPC

NETWORK FLOWS

Node 1: $x_{12} - x_{61} = 10$

Node 2: $x_{23} + x_{24} + x_{25} + x_{26} - x_{12} = -12$





$$\sum_{(i,j)\in E(i)} x_{ij} - \sum_{(j,i)\in I(i)} x_{ij} = \mathbf{b}_i \text{ , } i \in \mathrm{N}$$

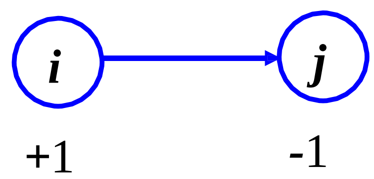
Notice that, for a feasible problem, $\sum_{i\in \mathrm{N}} \mathbf{b}_i = 0$

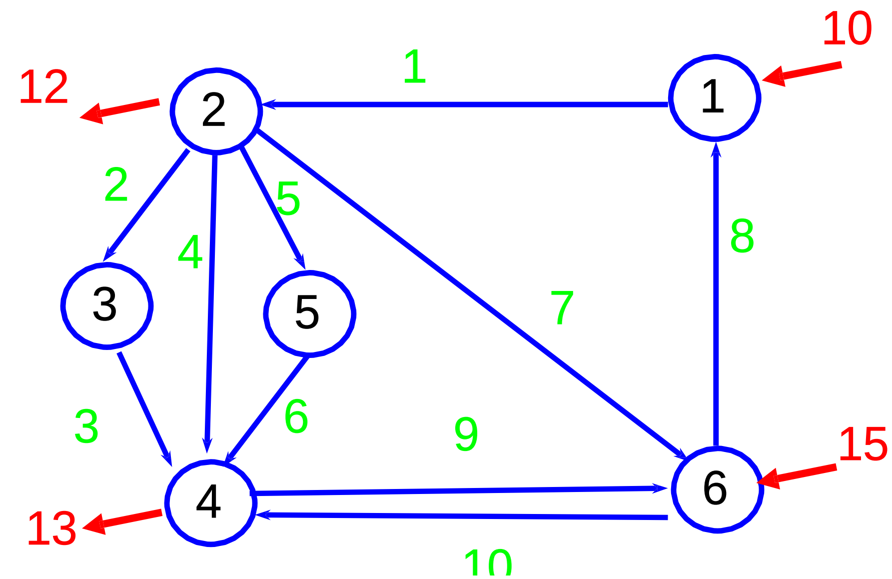
NETWORK FLOWS

Nudo 1: $x_1 - x_8 = 10$

Nudo 2: $x_2 + x_4 + x_5 + x_7 - x_1 = -12$

$\ldots$





$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ -1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & -1 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_9 \\ x_{10} \end{pmatrix} = \begin{pmatrix} 10 \\ -12 \\ 0 \\ -13 \\ 0 \\ 15 \end{pmatrix}$$

NODE-LINK INCIDENCE MATRIX *D*

FLOW VECTOR *x*

INP./OUTP. *b*

$x_i \geq 0,\ i = 1, \ldots, 10$

Sometimes upper bounds are required on the flows: $x_i \leq u_i$

MIN-COST FLOW PROBLEM: DEFINITION

D - **NODE-LINK INCIDENCE MATRIX**

$\boldsymbol{x}$ - **FLOW VECTOR (**DECISION VARIABLES**)**

b - **INJECTIONS/EXTRACTIONS VECTOR**

$\boldsymbol{l}, \boldsymbol{u}$ – **LOWER-UPPER BOUNDS VECTORS**

$$\text{Min}_{x}\ \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}$$
$$\boldsymbol{D}\boldsymbol{x} = \boldsymbol{b}$$
$$\boldsymbol{l} \leq \boldsymbol{x} \leq \boldsymbol{u}$$

Typically $\boldsymbol{l} = \boldsymbol{0}$;

if $\boldsymbol{l} \neq 0$ the problema can be easily reformulated using new decision variables $\boldsymbol{y} = \boldsymbol{x} - \boldsymbol{l}$

Exercise: reformulate the problem with the new variables y

Typically $\boldsymbol{c} \geq \boldsymbol{0}$;

In this case the Solution set F^* is bounded:

There exists r in IR, so that any solution x^* verifies $|x^*| \leq r$

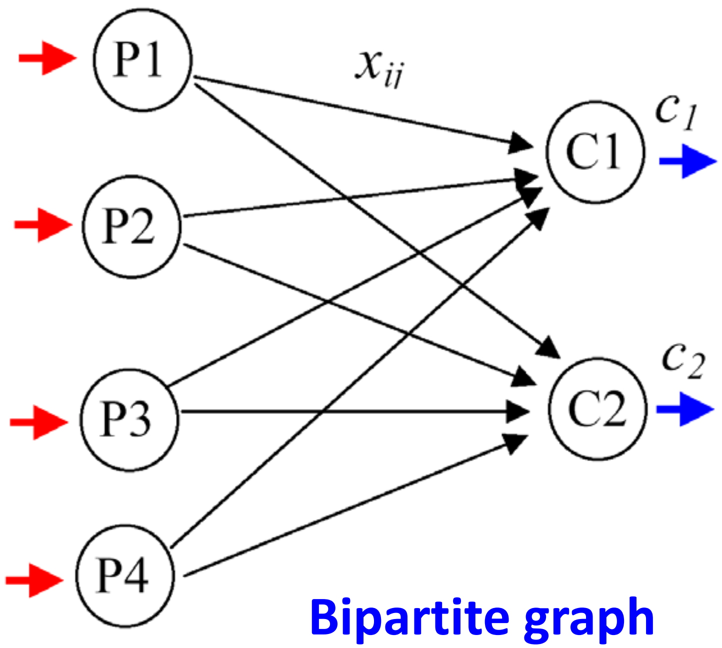
Exercise: find one such r

$$Min_x \sum_{i=1}^{n_p} \sum_{j=1}^{n_c} t_{i,j} x_{i,j}$$

$$\sum_{j=1}^{n_c} x_{ij} = p_i \; i = 1, 2, \ldots, n_p$$

$$\sum_{i=1}^{n_p} x_{ij} = c_j ; \; j = 1, 2, \ldots, n_c$$

$$x_{i,j} \geq 0$$



Bipartite graph

BALANCED PROBLEM

$$\sum_{j=1}^{n_p} p_j = \sum_{i=1}^{n_c} c_i$$

See more examples solved in the Problems and Lab Sessions

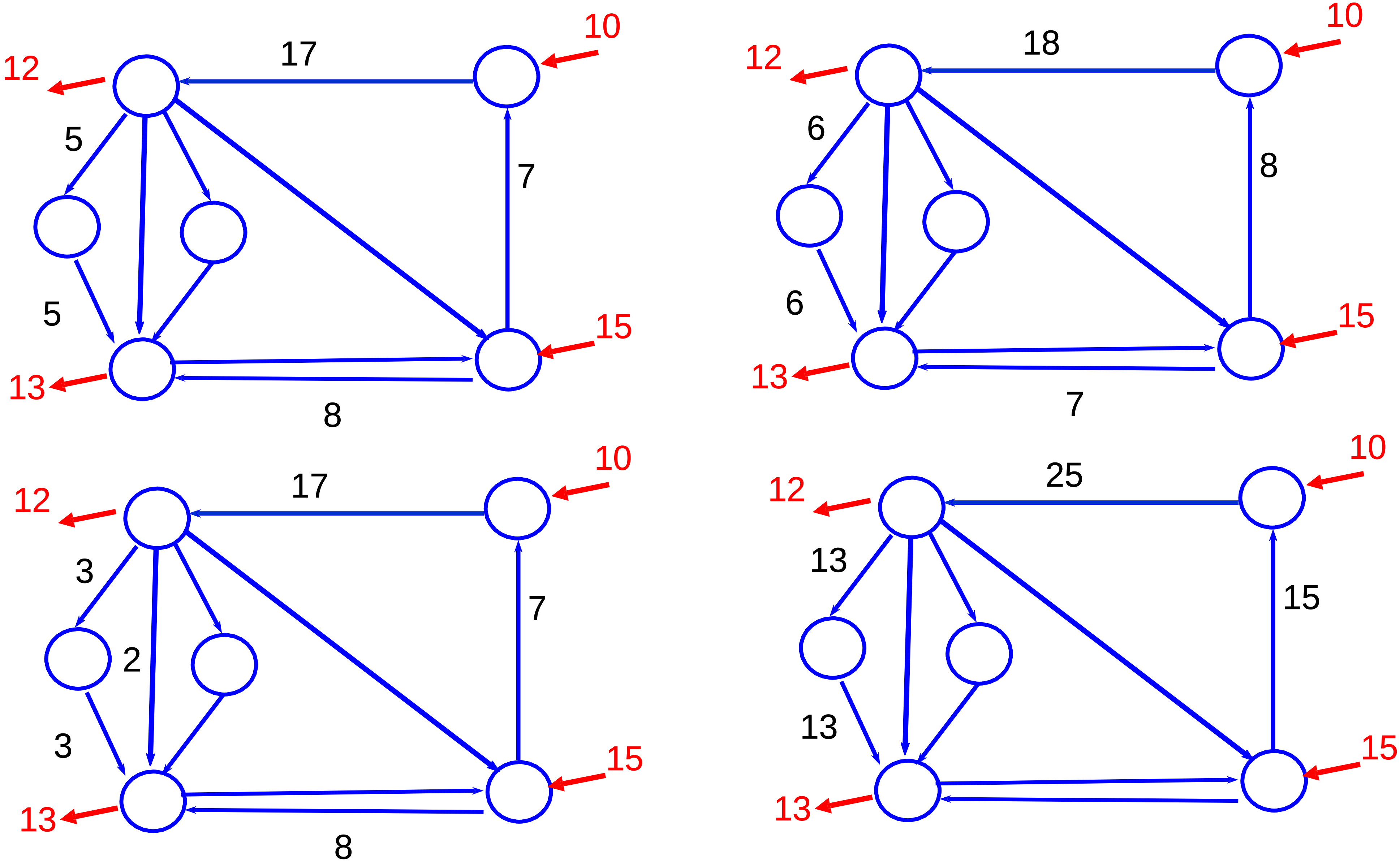
AMPL LANGUAGE: DECLARATIONS NODE, ARC

```
set CIUDADES;
set ARCOS within (CIUDADES cross CIUDADES);

param oferta {CIUDADES} >= 0;    # injections
param demanda {CIUDADES} >= 0;   # extractions

param coste {ARCOS} >= 0; # costs  of transp.

minimize Total_Coste;

node Nodo {k in CIUDADES}: net_in=demanda[k]-oferta[k];

arc enlace {(i,j) in ARCOS} >= 0,
  from Nodo[i], to Nodo[j], obj Total_Coste coste[i,j];
```

```
check: sum {i in CIUDADES}

       oferta[i] = sum {j  in CIUDADES}    demanda[j];
```

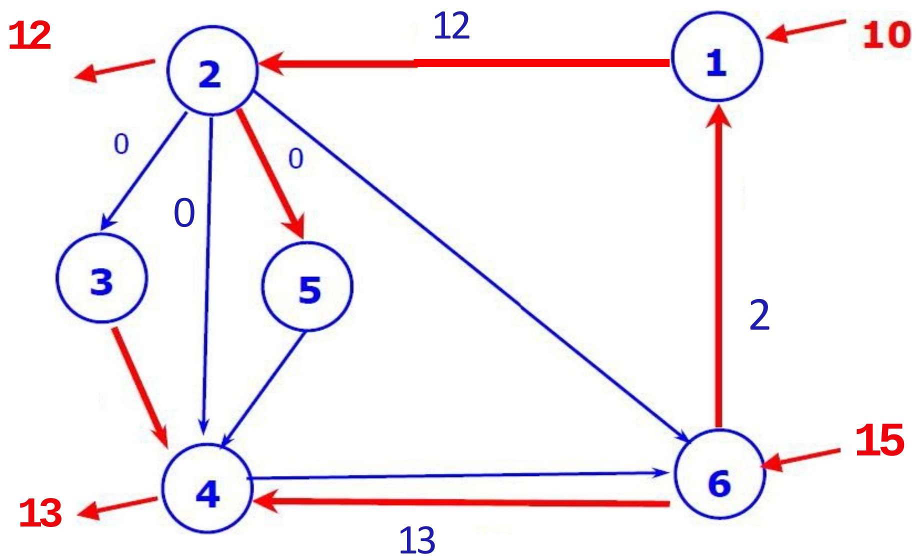
CONTENTS

- PROBLEM DEFINITION
- **CONCEPT OF BASIC FEASIBLE SOLUTION (bfs)**
  - Definition
  - Sets of indexes (basic, non-basic)
  - Why basic feasible solutions?
- OUTLINE OF THE ALGORITHM
- THE ALGORITHM IN DETAIL
  - Testing for optimality
  - Choosing a better bfs
  - Formal definition of the algorithm



Examples of feasible flows

BASIC FEASIBLE FLOWS



Some feasible flow vectors are specially relevant for the simplex algorithm.
We will refer to them as ***basic*** feasible flows

The components x_{ij} (indexes (i,j)) of a basic feasible flow vector x can be regrouped into 2 (3) sets of components (indexes)

	Type of Problem	
	No upper bounds	With upper bounds
Basic set	I_{B}	I_{B}
Non-basic set(s)	I_{N}	$I_{\mathrm{N+}}$, $I_{\mathrm{N-}}$

IDENTIFYING A BASIC FEASIBLE FLOW

Let $G = (N, A)$. Let $x_{i,j},\ (i,j) \in A$ be a feasible flow.

$x_{i,j},\ (i,j) \in A$ is basic feasible flow

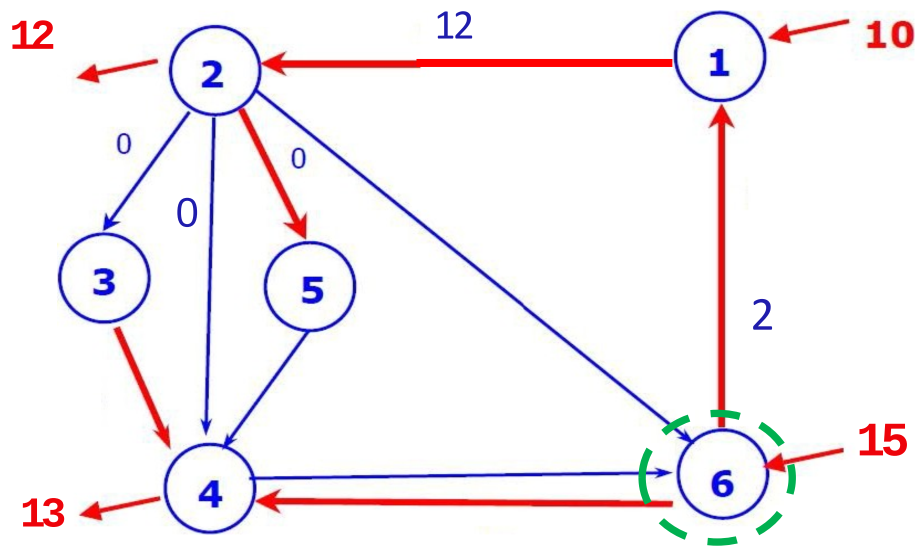
0. It is feasible (i.e., verifies the problem constraints)

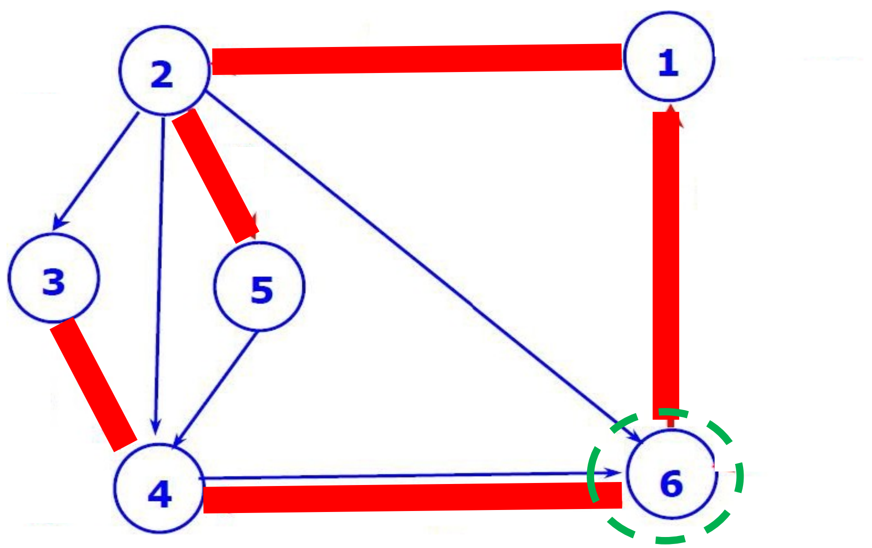
1. $|I_B| = |N| - 1$
   $(i,j) \in I_B \longrightarrow x_{i,j} = 0,\ x_{i,j} = u_{i,j}$ or $0 < x_{i,j} < u_{i,j}$
2. Links in I_B build up a Spanning Tree
3. $(i,j) \in I_{N-} \longrightarrow x_{i,j} = 0$
4. $(i,j) \in I_{N+} \longrightarrow x_{i,j} = u_{i,j}$

$I_{N^+} = \{\, a \notin I_B, \,|\, x_a = u_a \,\}$ $\qquad$ $I_{N^-} = \{\, a \notin I_B, \,|\, x_a = 0 \,\}$

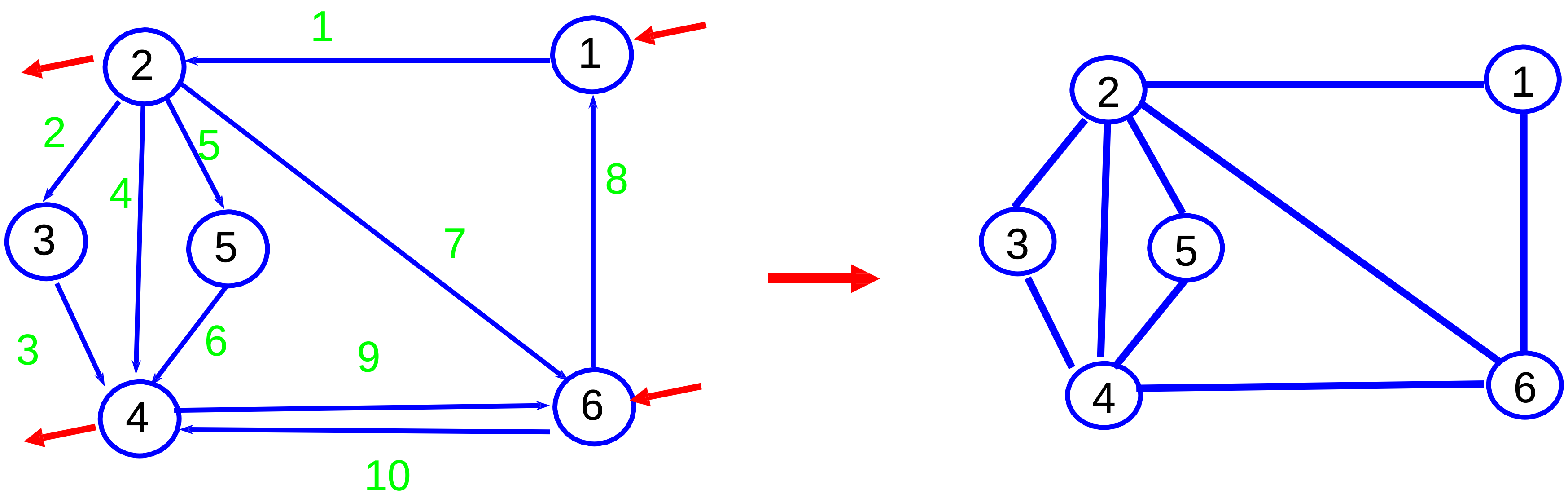
BUILDING UP A BASIC SET

- **The basic set of a basic feasible solution is made up by $|N|$-1 links;**
- **The set of asocciaated undirected links must form a spanning tree.**
- **One of the nodes is arbitrarily chosen as the root node**
- **A spanning tree is a subgraph of the original graph containing no cycles**
- **Notice that any node can be accessed by the root node through the spanning tree**

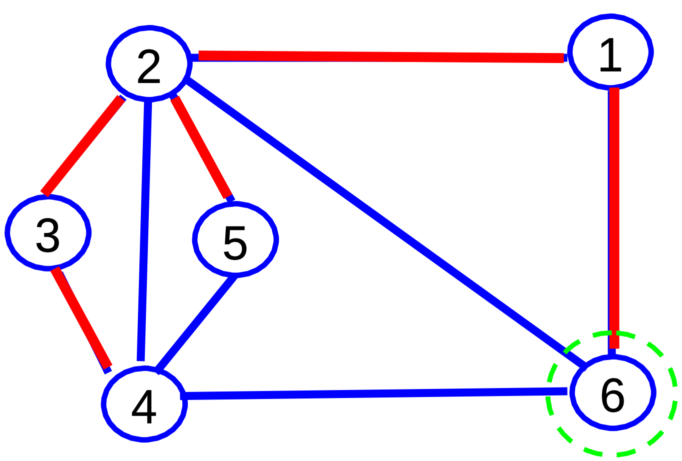




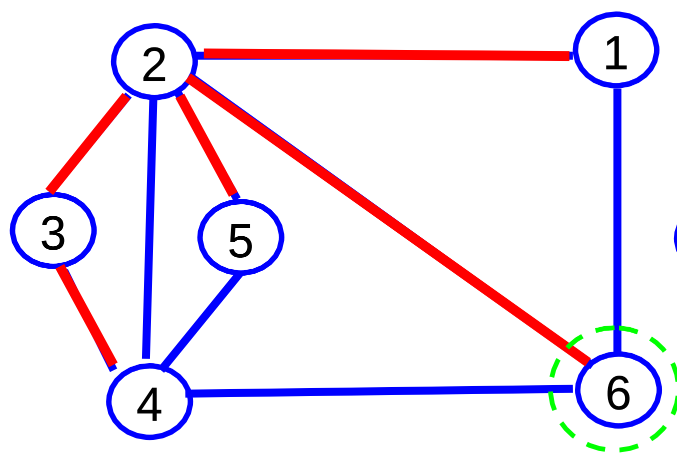
MORE EXAMPLES.



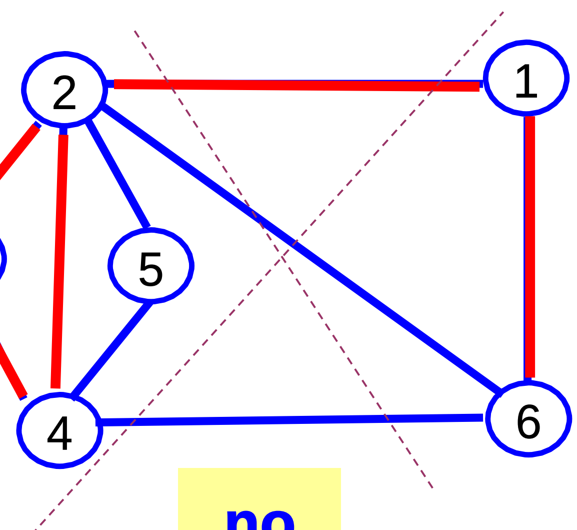
SPANNING TREE: m-1 links; no cycles; select a node as the root of the tree.



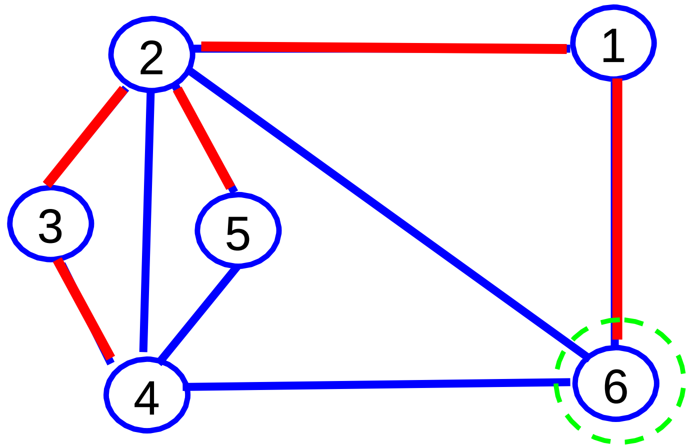
yes

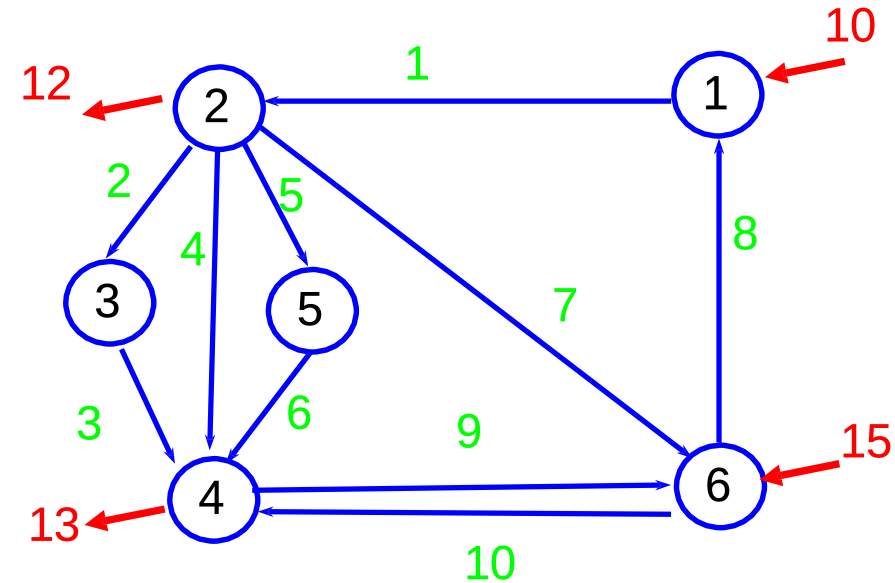


yes



no



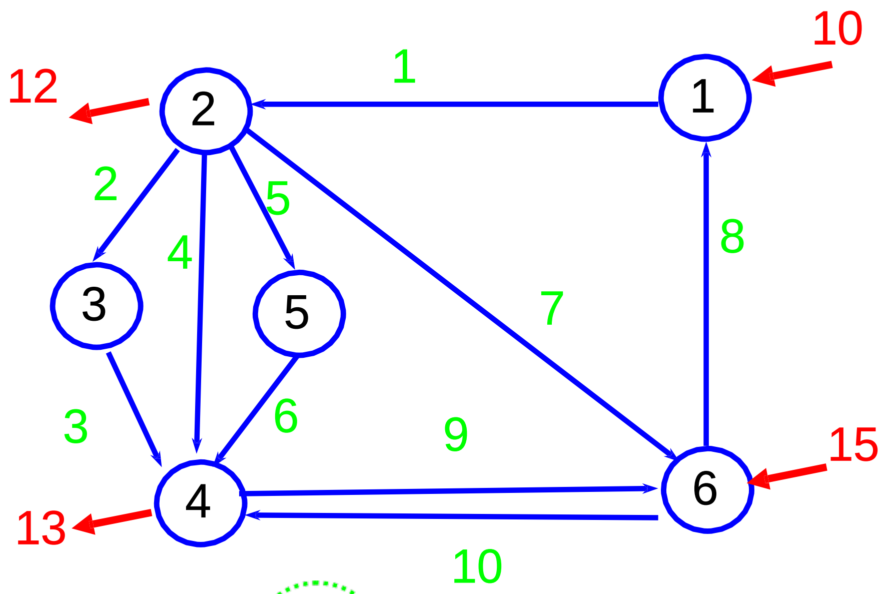


$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ -1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & -1 & 1 \end{pmatrix}$$

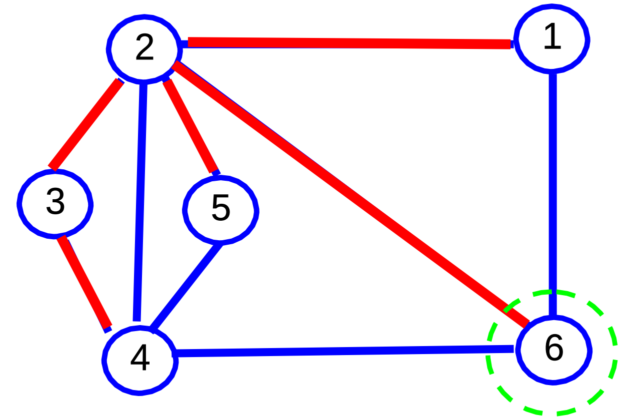
$$\begin{pmatrix} 1 & 0 & 0 & 0 & -1 \\ -1 & 1 & 0 & 1 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ \cancel{0} & \cancel{0} & \cancel{0} & \cancel{0} & \cancel{1} \end{pmatrix} \rightarrow B = \begin{pmatrix} 1 & 0 & 0 & 0 & -1 \\ -1 & 1 & 0 & 1 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \end{pmatrix}$$

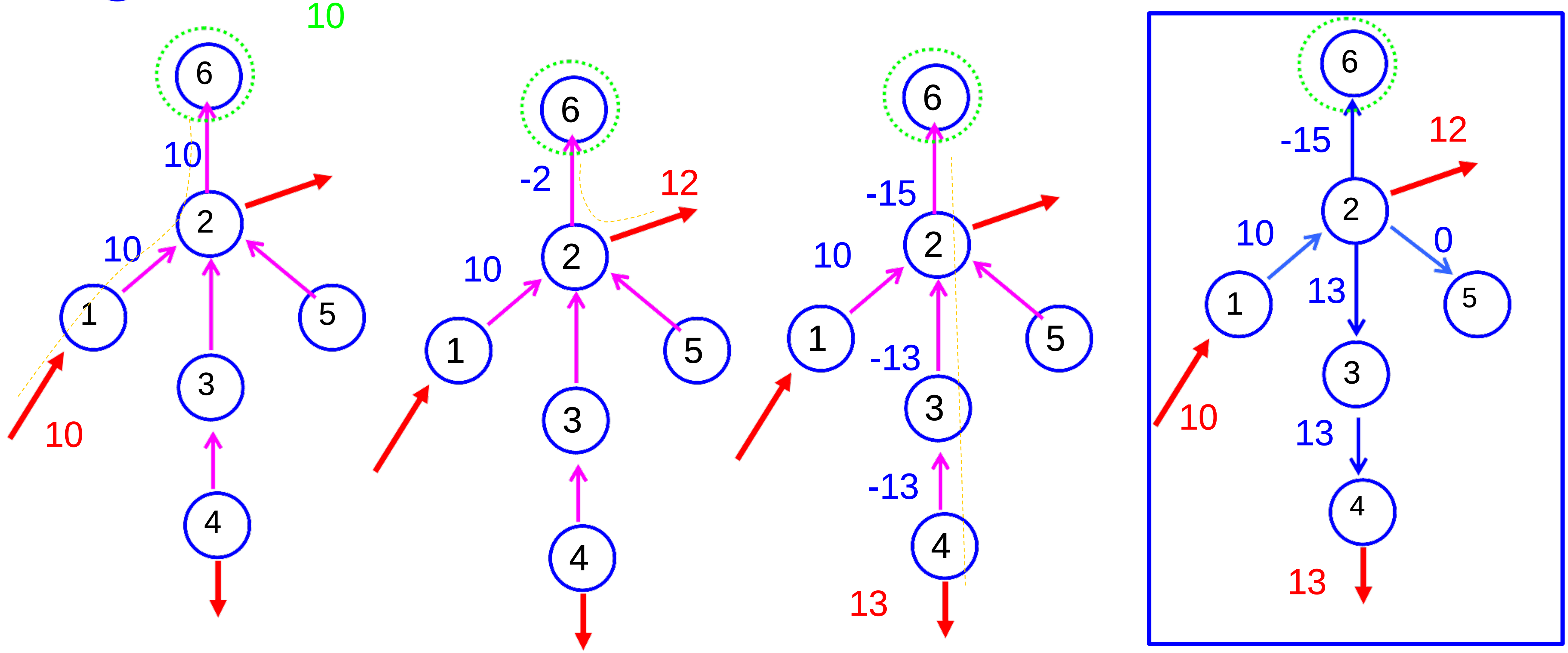
A Basic Solution formed by this procedure (building a spanning tree) is called a basic feasible solution (bfs) if the following condition is met:

$$B^{-1}\, b \geq 0$$



NON-FEASIBLE BASIS





PROBLEMS WITH NO UPPER BOUNDS: CHECKING THE FEASIBILITY

Why basic feasible solutions?

FUNDAMENTAL THEOREM OF LINEAR PROGRAMMING.
IF A LINEAR PROGRAM ACHIEVES A FINITE SOLUTION (in other words, the optimal objective function value is finite),

↓

THEN THERE EXISTS, AT LEAST, A BASIC FEASIBLE SOLUTION WHERE THE OBJECTIVE FUNCTION ACHIEVES ITS OPTIMAL VALUE (in other words, these basic feasible solutions are optimal).

Because of that it makes sense to look for a bfs

AND

identify a method that, from a bfs, gets another one with better objective function (and repeat until no further enhancement is possible)

CONTENTS

OUTLINE OF THE ALGORITHM

0) Find an initial bfs
1) Evaluate the obj.f. on the current bfs
2) TEST: is the current bfs optimal?
    1) YES: STOP
    2) NO: Find a better bfs
3) GOTO 1

UPC

CONTENTS

- PROBLEM DEFINITION
- CONCEPT OF BASIC FEASIBLE SOLUTION (bsf)
  - Definition
  - Sets of indexes (basic, non-basic)
  - Why basic feasible solutions?
- OUTLINE OF THE ALGORITHM
- **THE ALGORITHM IN DETAIL**
  - Testing for optimality:
    - dual variables; reduced costs; optimality conditons
  - Choosing a better bfs
    - Problems with upper bounds.
      The case $I_{N+} \leftrightarrow I_{N-}$ The case $I_{N+} \leftrightarrow I_{B}$
  - Formal definition of the algorithm

UPC

DUAL VARIABLES ASSOCIATED TO AN INDEX SET I_B

MEANING AND INTERPRETATION OF DUAL VARIABLES IN THE MIN-COST FLOW PROBLEM

Once a basic feasible solution is determined,
(or equivalently, a Spanning Tree and a root node has been determined)

- The dual variable $\boldsymbol{\lambda}_i$ for a node i is the '- cost' to reach that node i from the root node, using the paths marked by the spanning tree.
  (In order to adjust to standard mathematical formulations the - is adopted)
- By convention, the dual variable $\boldsymbol{\lambda}_r$ of the root node r is set to $\boldsymbol{\lambda}_r = 0$

- **Dual variables are associated to the structure marked by I_B**

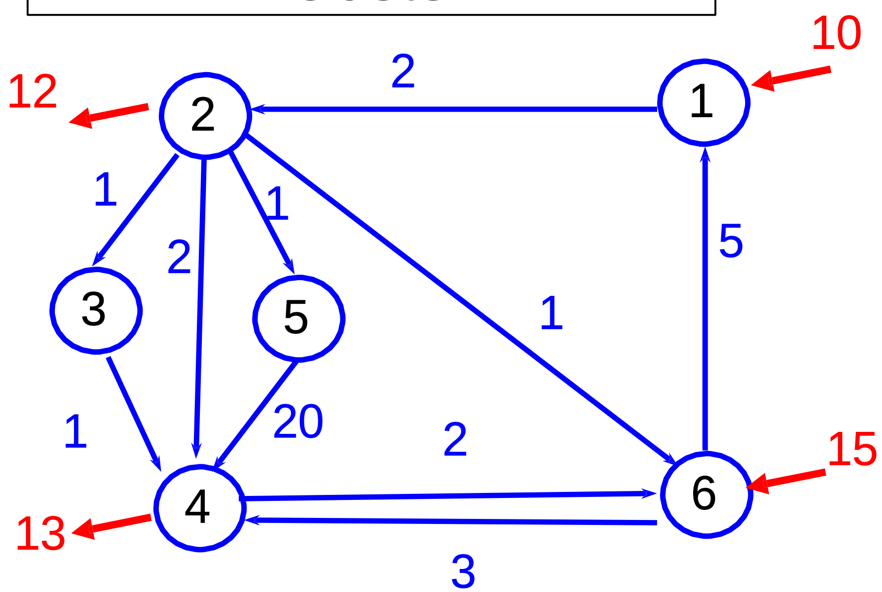
  If another basic feasible solution is considered, dual variables are different.

UPC

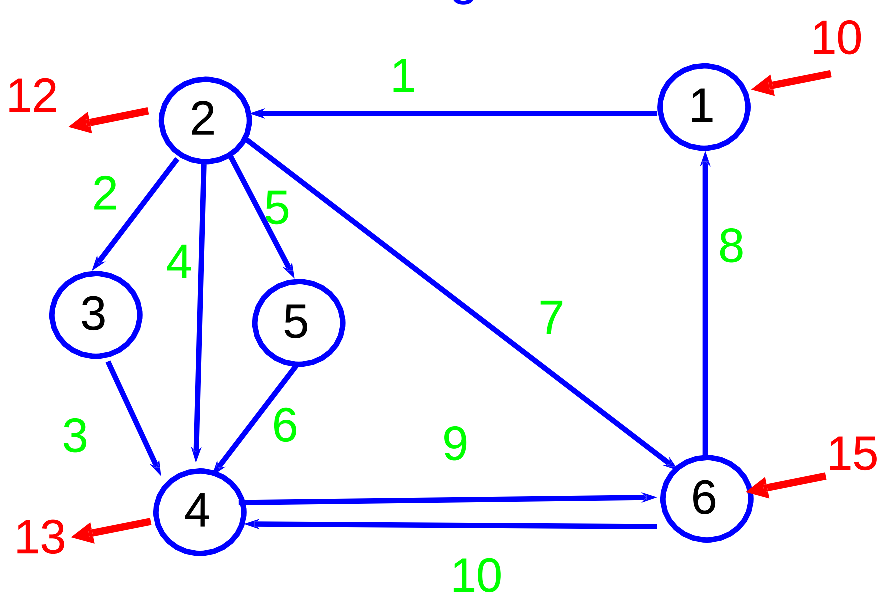
DUAL VARIABLES ASSOCIATED TO AN INDEX SET I_B

$$c^\top = (2, 1, 1, 2, 1, 20, 1, 5, 2, 3)$$

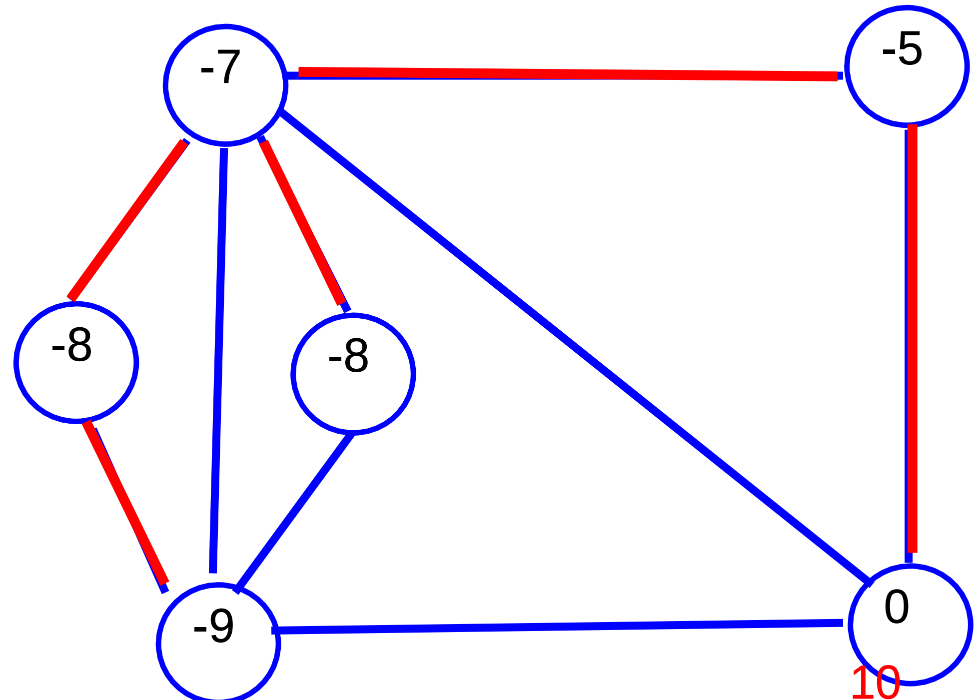
Link Costs

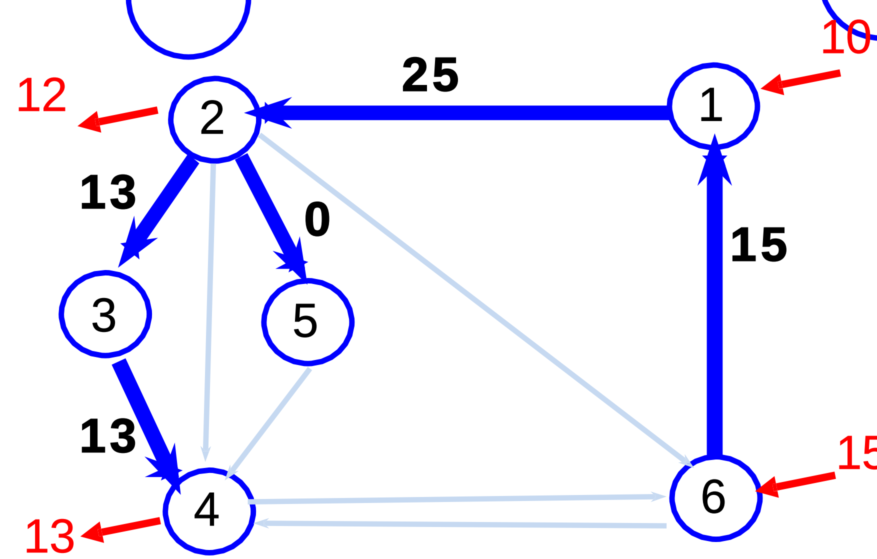


Link Number

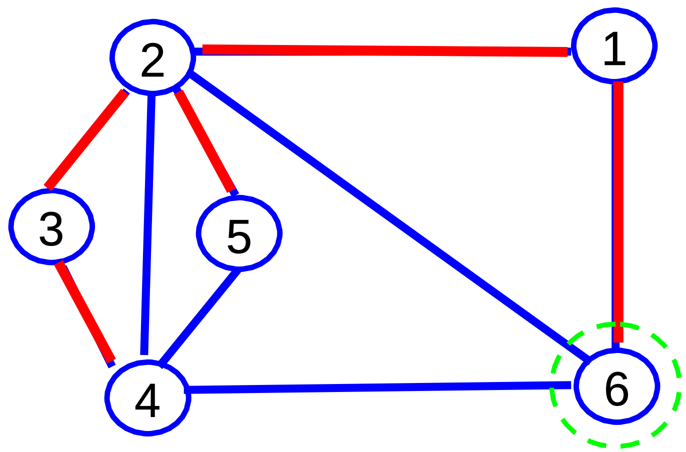


Dual variables using the Spanning Tree



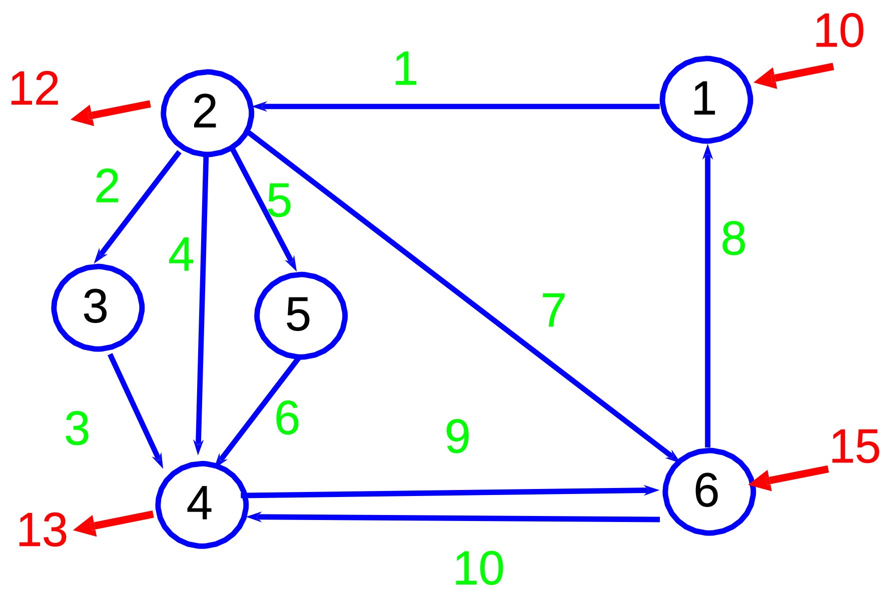


DUAL VARIABLES ASSOCIATED TO AN INDEX SET I_B



$$B^\top \lambda = c_B$$

$$\begin{pmatrix} -1 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & 1 & 0 & 0 & -1 \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \\ \lambda_5 \end{pmatrix} = \begin{pmatrix} c_{61} \\ c_{12} \\ c_{23} \\ c_{34} \\ c_{25} \end{pmatrix}$$



Feasible basis

$$\lambda_1 = -c_{61}$$
$$\lambda_2 = -c_{61} - c_{12}$$
$$\lambda_3 = -c_{61} - c_{12} - c_{23}$$
$$\lambda_5 = -c_{61} - c_{12} - c_{25}$$
$$\lambda_4 = -c_{61} - c_{12} - c_{23} - c_{34}$$
$$(\lambda_6 = 0)$$

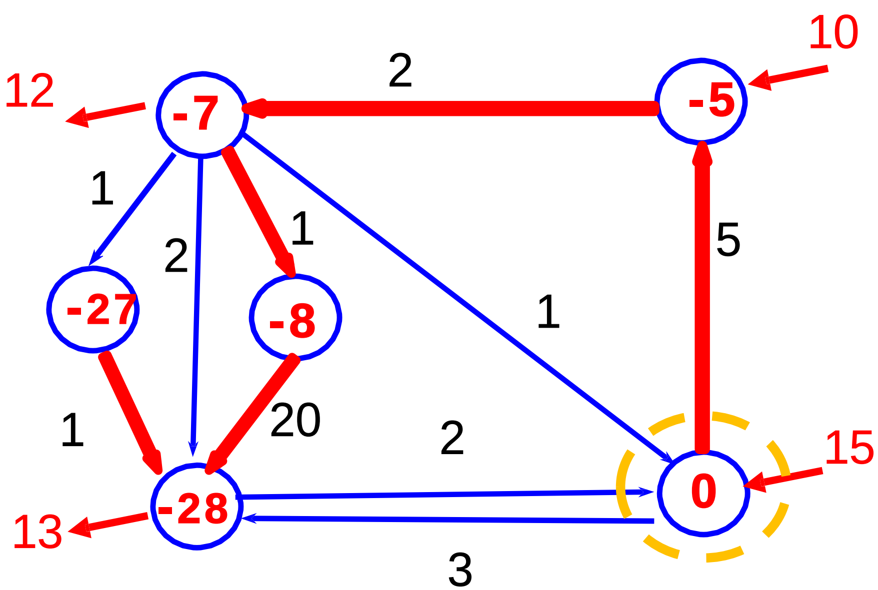
λ_i = **−** Unit cost from root $\to$ ***i***

Algebraic calculation

REDUCED COSTS ASSOCIATED TO AN INDEX SET I_B

Looking for a better BFS

Let us focus on node **4**. Currently, it is accessed through node 3 (using the Spanning Tree) with a cost of 9.
If link **(2,3)** is dropped from the current basis and link **(5,4)** is added to form a new basis …
Would things go better??

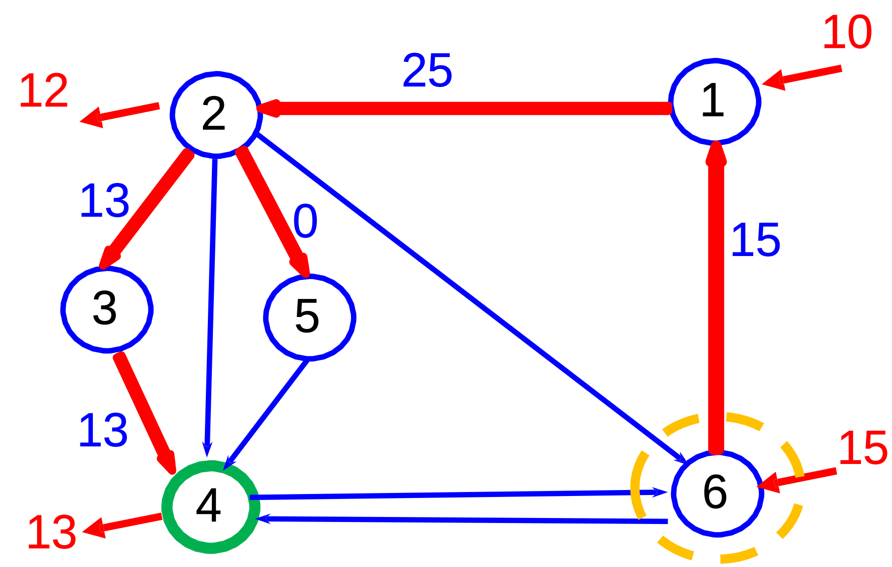


The decrease per unit of flow reaching node 4 would be:

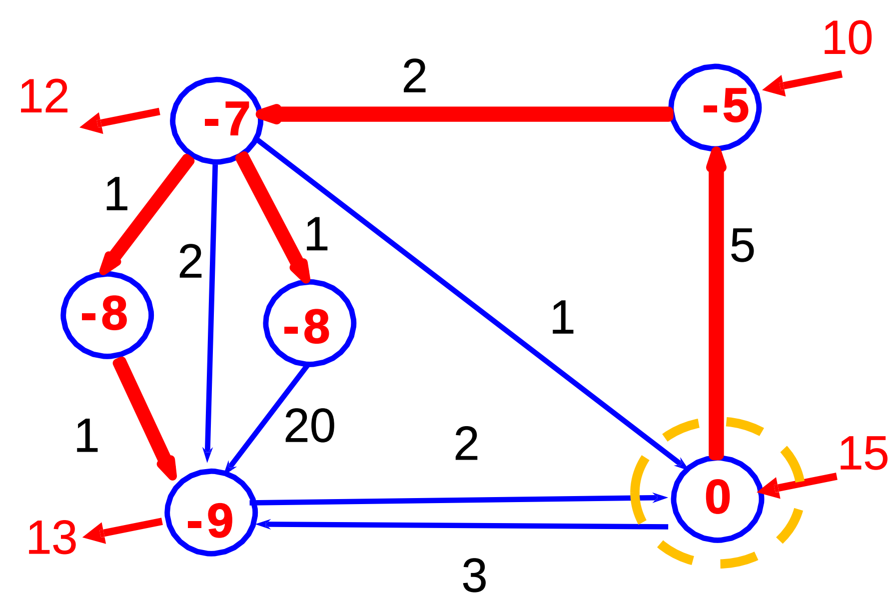
9 – 28 = -19

Clearly, it is a bad option!

Obj.Function value = 151



Link Flows

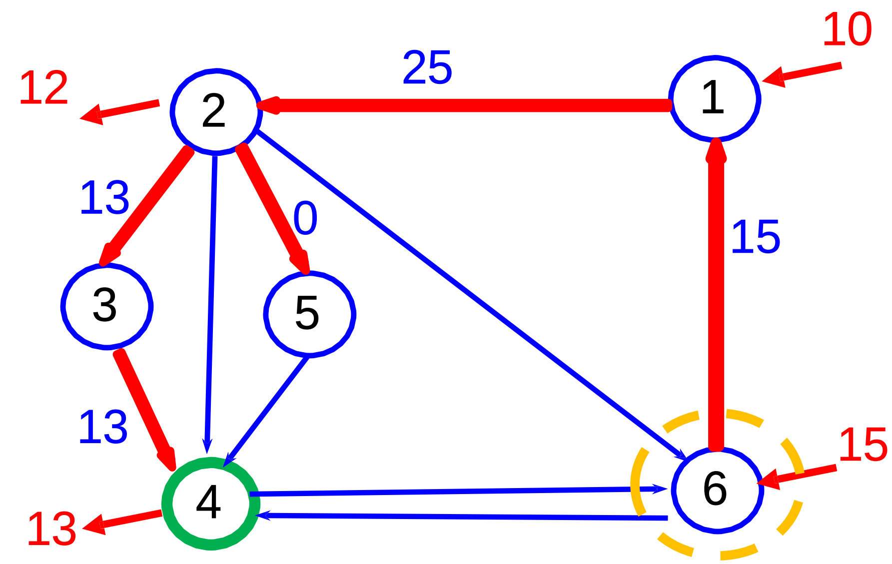


Link costs & dual variables

REDUCED COSTS ASSOCIATED TO AN INDEX SET I_B

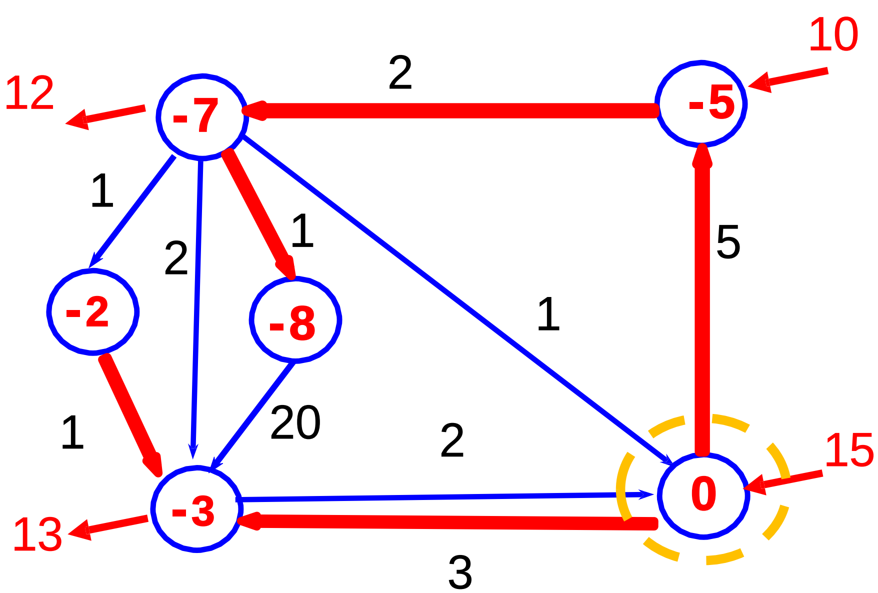
Looking for a better BFS

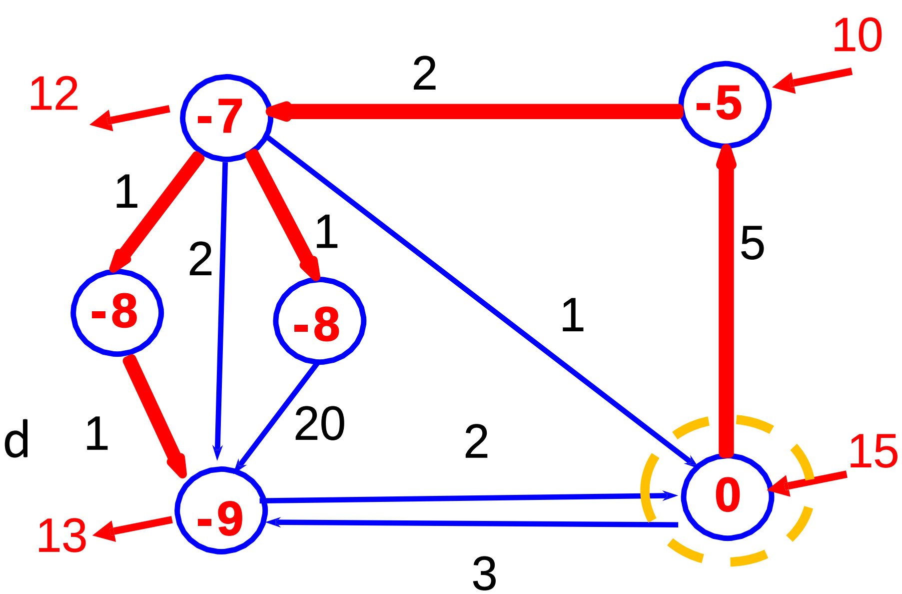
Obj.Function value = 151



Link Flows

Let us focus on node **4**. Currently, it is accessed through node 3 (using the Spanning Tree) with a cost of 9.
If link **(2,3)** is dropped from the current basis and link **(6,4)** is added to form a new basis …
Would things go better??





Link costs & dual variables

The decrease per unit of flow reaching node 4 would be: **9 – 3 = 6**

Clearly, it is a good option!

(Provided that flows can be accommodated on the new basis)

REDUCED COSTS ASSOCIATED TO AN INDEX SET I_B

Reduced costs calculation

Let link $a_k = (i, j),\ k \in I_N$:

$$r_k = c_k - (\lambda_i - \lambda_j)$$

$$c^\top = (2, 1, 1, 2, 1, 20, 1, 5, 2, 3)$$

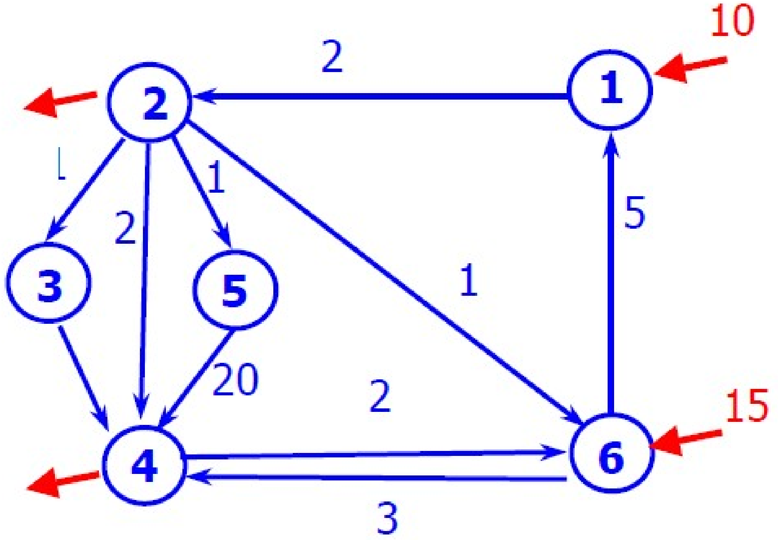
$$r_{64} = c_{64} - (\lambda_6 - \lambda_4) = 3 - (0 + 9) = -6$$

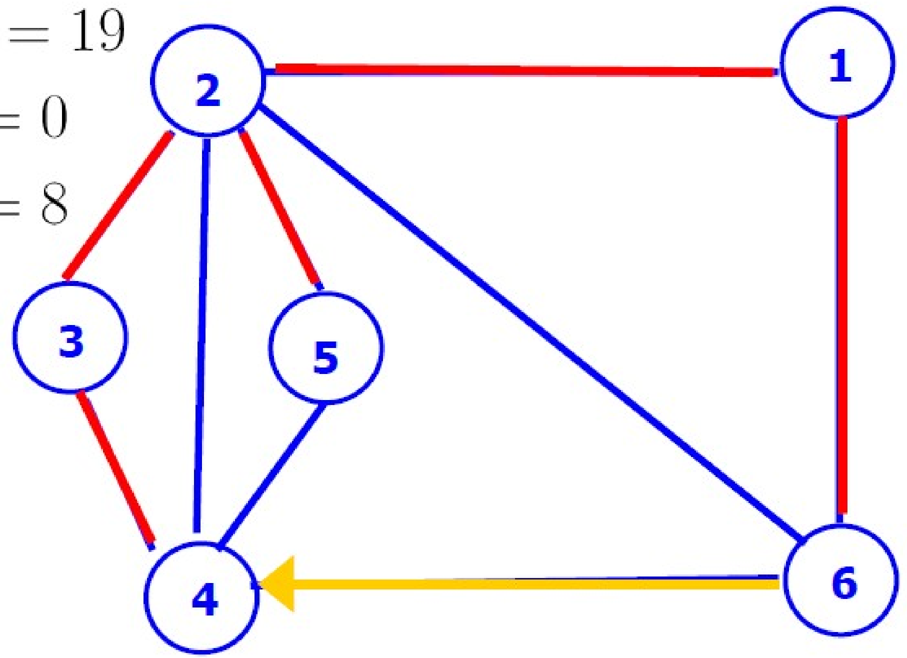
$$r_{46} = c_{46} - (\lambda_4 - \lambda_6) = 2 - (-9 - 0) = 11$$

$$r_{54} = c_{54} - (\lambda_5 - \lambda_4) = 20 - (-8 + 9) = 19$$

$$r_{24} = c_{24} - (\lambda_2 - \lambda_4) = 2 - (-7 + 9) = 0$$

$$r_{26} = c_{26} - (\lambda_2 - \lambda_6) = 1 - (-7 + 0) = 8$$





DETERMINING AN ENTERING NON-BASIC VARIABLE

CONDITIONS FOR AN OPTIMAL INDEX SET I_B

Let I_B a set of feasible basic indices associated with matrix B.

$$I_{N^+} = \{\, a \notin I_B, \,|\, x_a = u_a \,\}$$

$$I_{N^-} = \{\, a \notin I_B, \,|\, x_a = 0 \,\}$$

An optimal basic solution is characterized by:

- If $a \in I_B$ then $r_a = 0$
- If $a \in I_{N^+}$ then $r_a \leq 0$ ← **Why this condition??**
- If $a \in I_{N^-}$ then $r_a \geq 0$

CONDITIONS FOR A *UNIQUE* SOLUTION OF THE PROBLEM

Let I_B a set of feasible basic indices associated with matrix B.

$$I_{N^+} = \{ a \notin I_B, | x_a = u_a \}$$

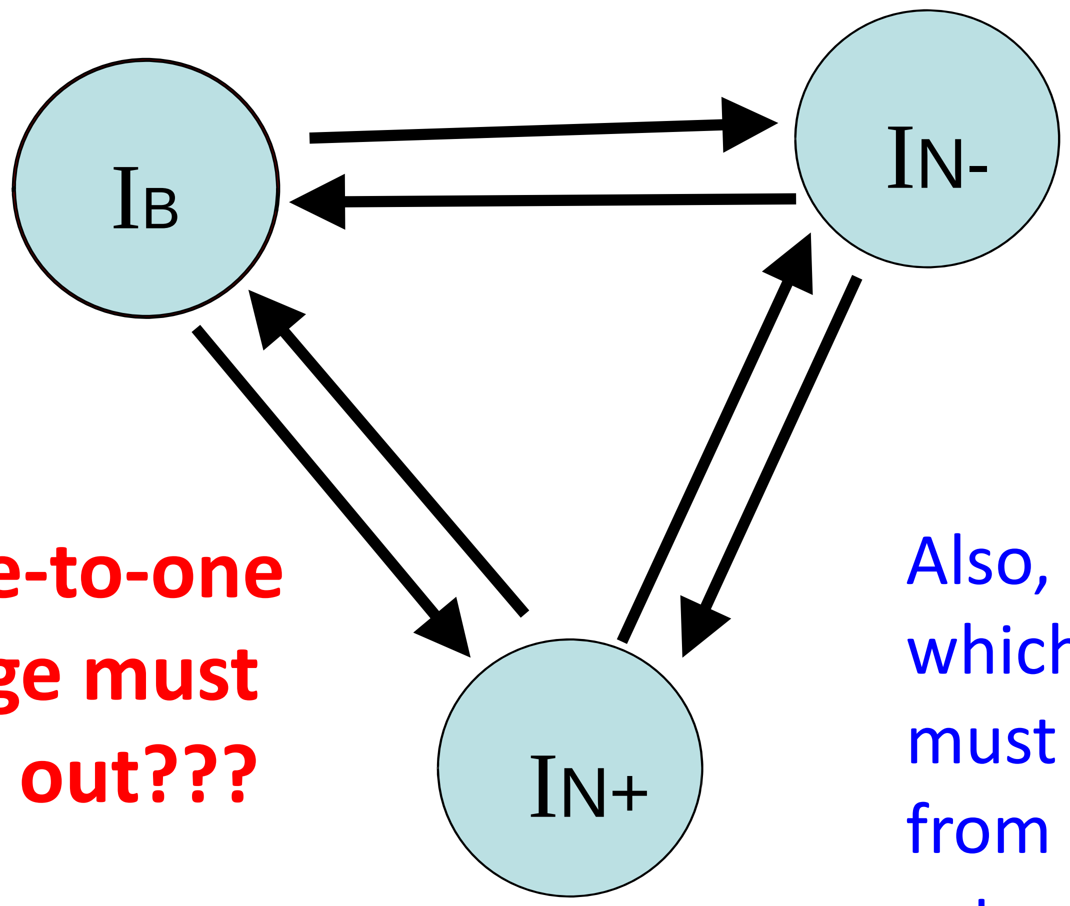
$$I_{N^-} = \{ a \notin I_B, | x_a = 0 \}$$

Then, if it verifies the following conditions, is the *unique* solution of the problem

- If $a \in I_B$ then $r_a = 0$
- If $a \in I_{N^+}$ then $r_a < 0$ ← **Why this condition??**
- If $a \in I_{N^-}$ then $r_a > 0$

By now we know how to identify indices $I_{B,}$ $I_{N+,}$ I_{N-} defining an optimal solution.

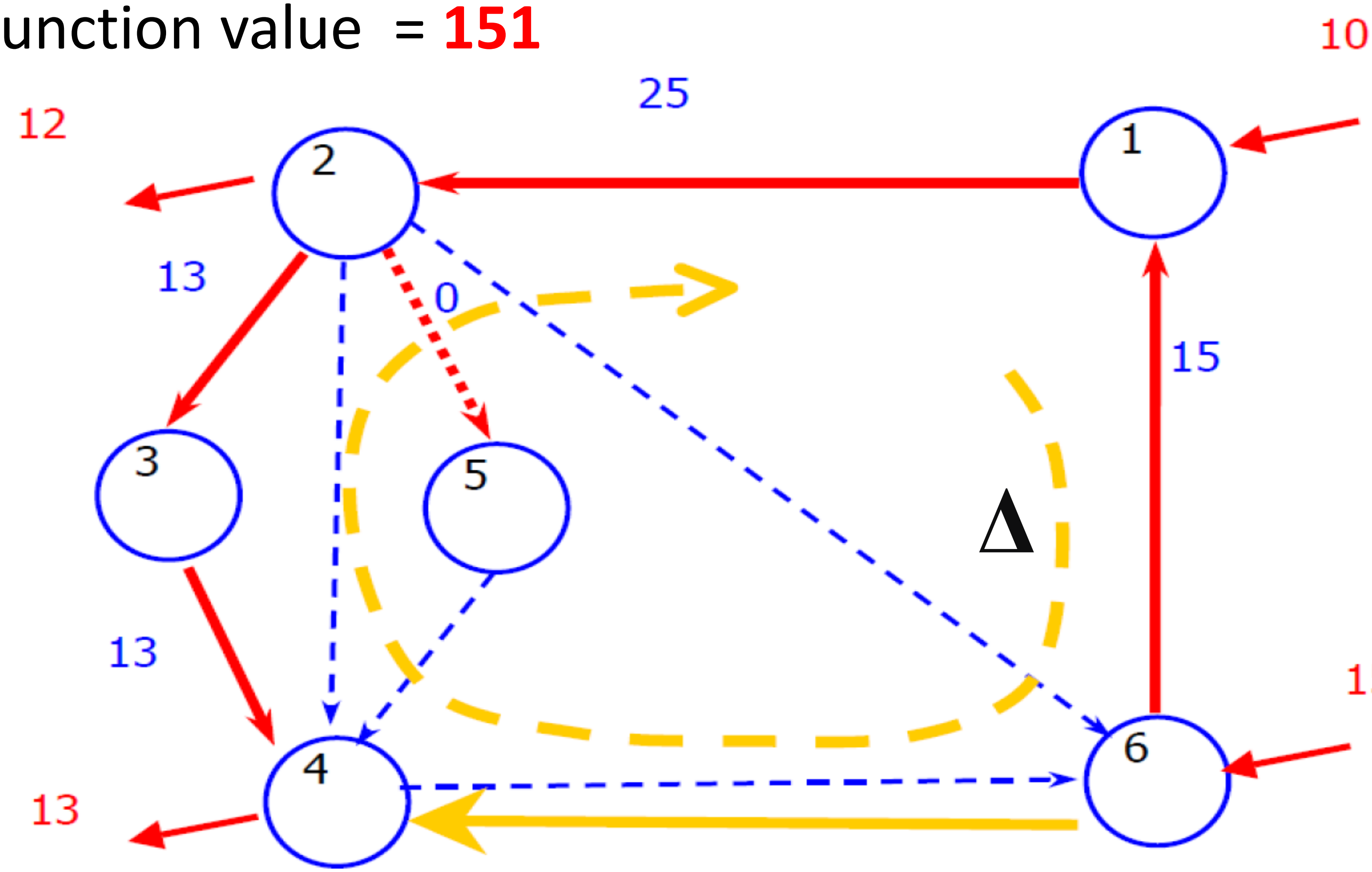
If the current solution is not optimal, the following one-to-one interchanges between sets $I_{B,}$ $I_{N+,}$ I_{N-} may occur in order to obtain new sets $I_{B,}$ $I_{N+,}$ I_{N-}



How a one-to-one interchange must be carried out???

Also, we know which links (i,j) must be chosen from the non-basic sets !!

Obj. Function value = **151**



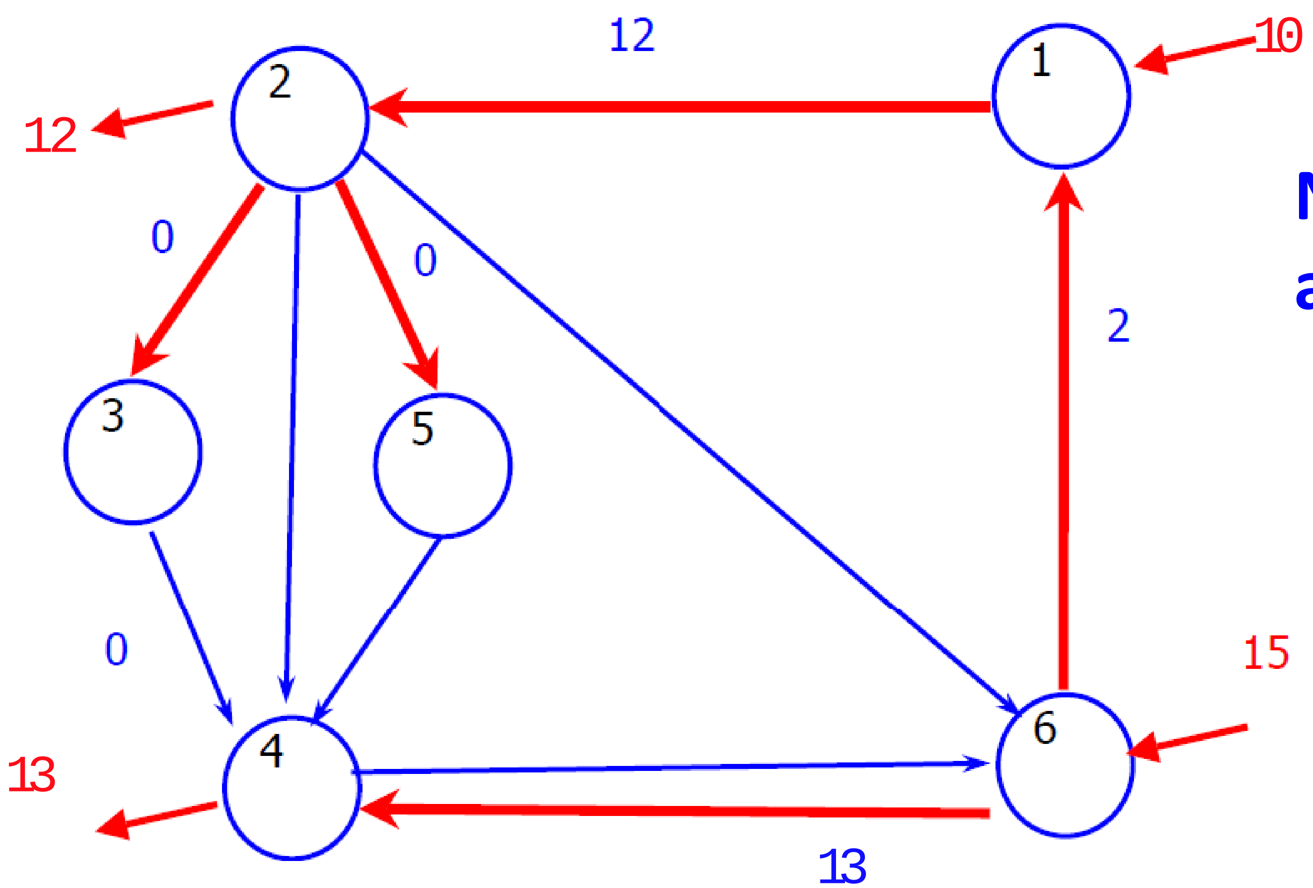
$$\Delta = x_{6,4} = Min\ \{\ 13, 13, 25, 15\ \} = 13$$

EXITING BASIC VARIABLE: (3,4) or (2,3) (Tie!)

Notice the sign (+) that will be applied to Δ !!!

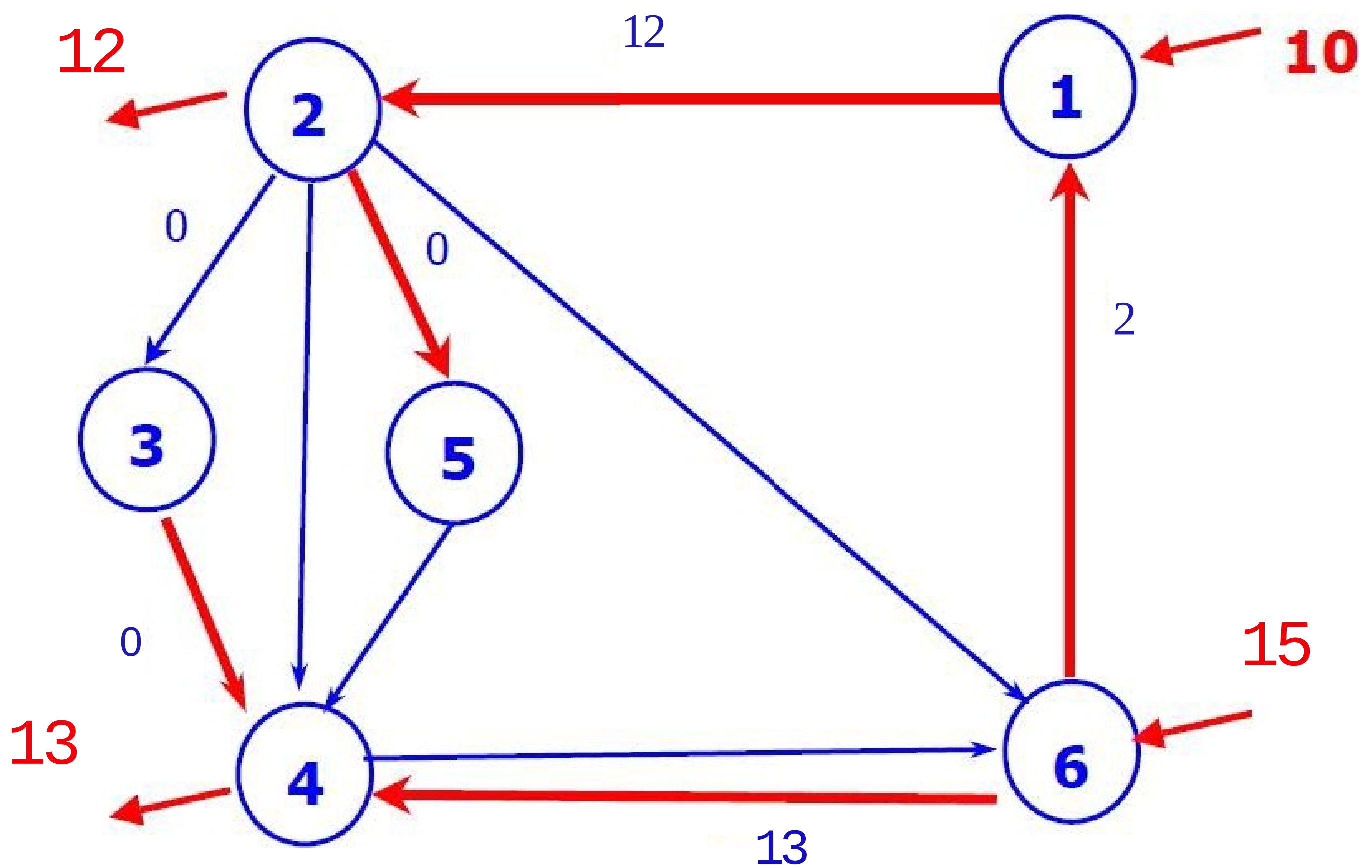
New Obj. Function value = 2x5 + 2x12 + 3x13 = **73**

New Obj. Function value =
= Previous value + (reduced cost r_{ij} of entering non-basic variable (*i,j*)) x (+Δ) =
= **151** + (-6)x13 = 151 – 78 = **73**



Notice the sign (+) applied to Δ !!!

NEW BASIC FEASIBLE SOLUTION

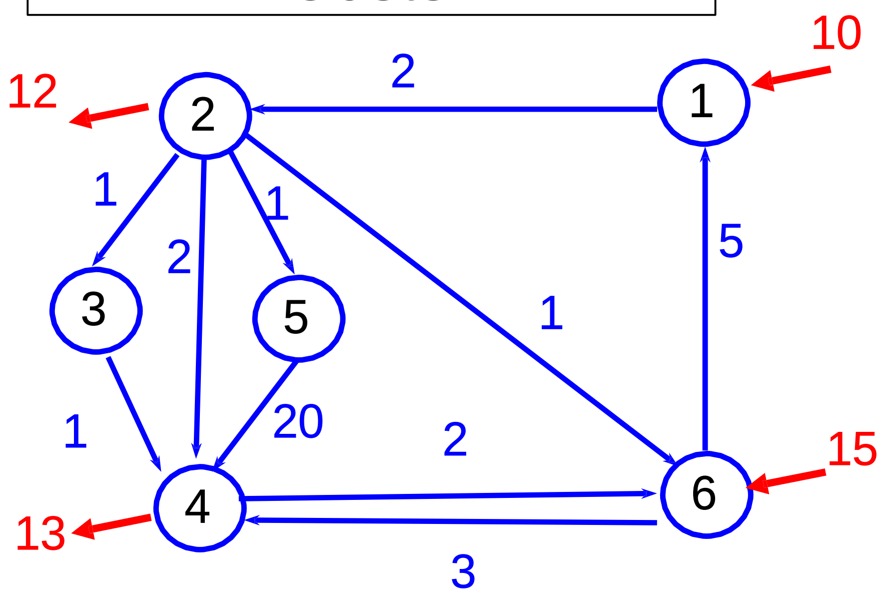


ALTERNATIVE BASIC SOLUTION

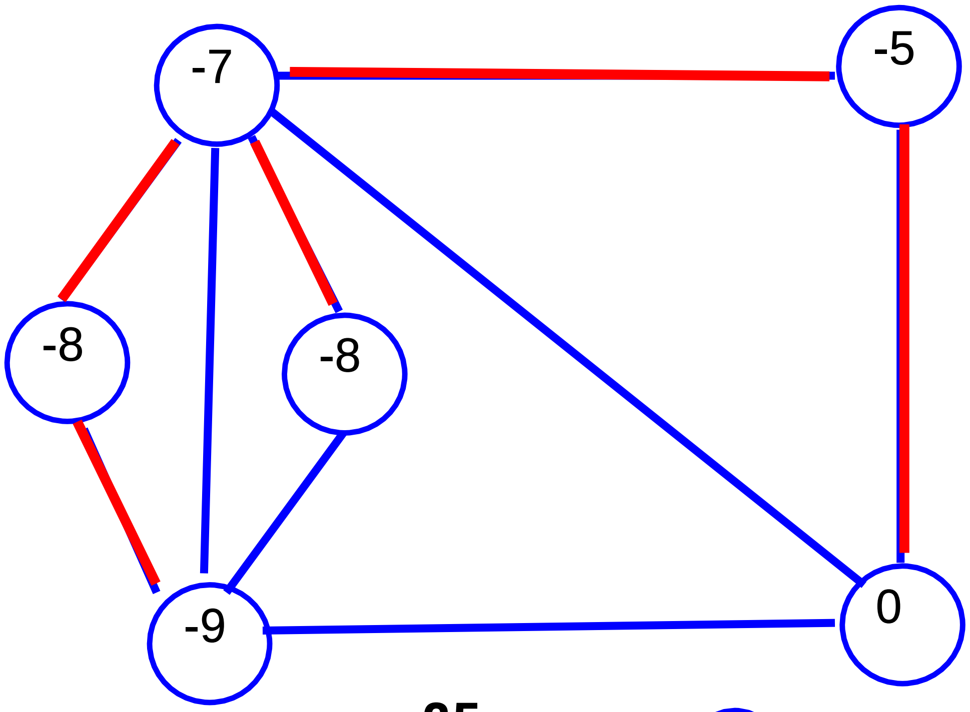
PROBLEMS with UPPER BOUNDS. $I_{N+} \leftrightarrow I_{N-}$

$$c^\top = (2, 1, 1, 2, 1, 20, 1, 5, 2, 3)$$

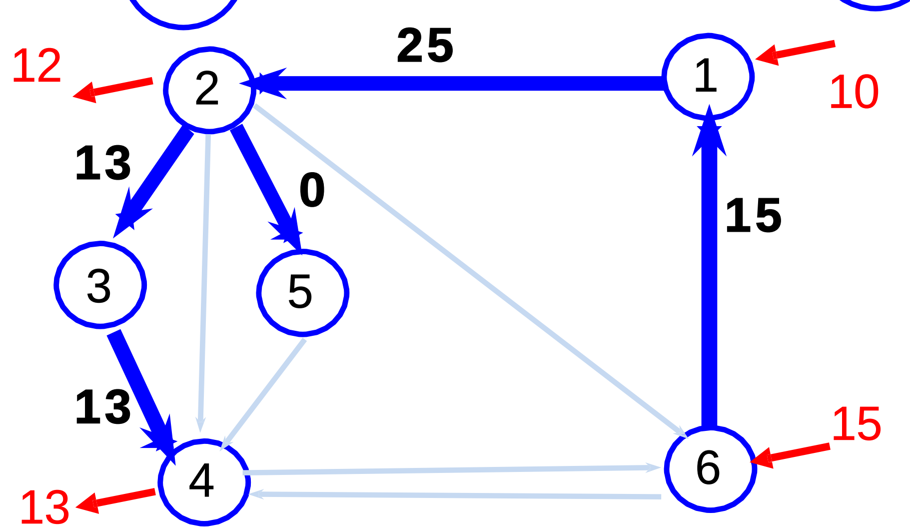
Link Costs



Dual variables using the Spanning Tree



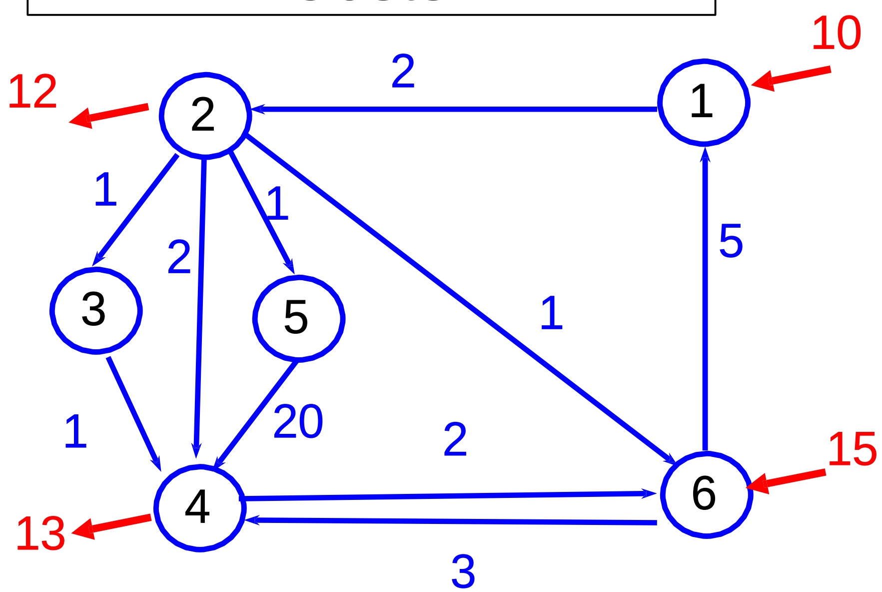
Assume now that $x_{64} \leq 10$ and try to find the optimal solution



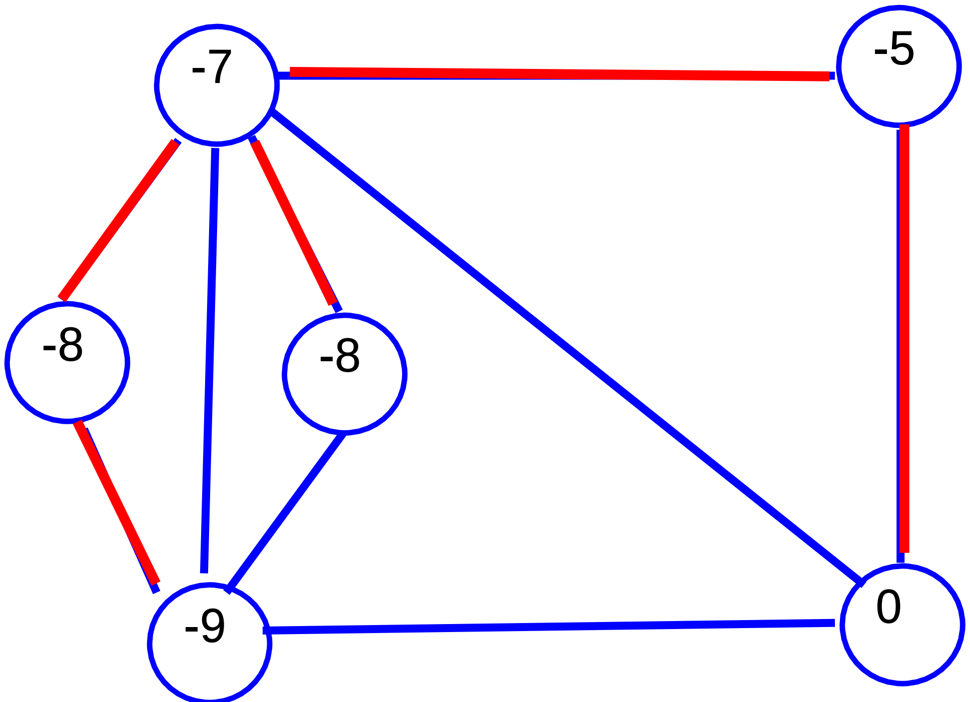
PROBLEMS with UPPER BOUNDS. $I_{N+} \leftrightarrow I_{N-}$

$$c^\top = (2,1,1,2,1,20,1,5,2,3)$$

Link Costs

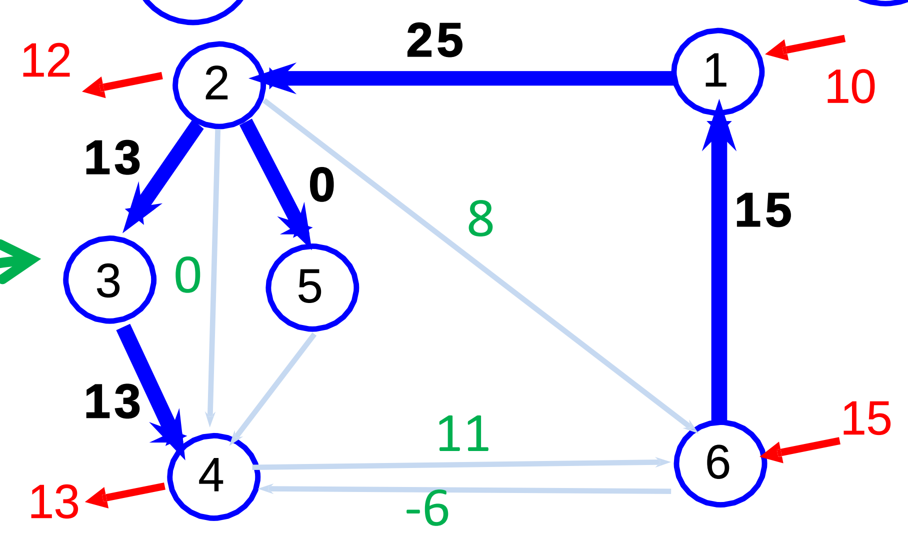


Dual variables using the Spanning Tree



I_B={ (6,1), (1,2), (2,3), (3,4), (2,5) }
I_{N+}= empty
I_{N-} = the remaining links

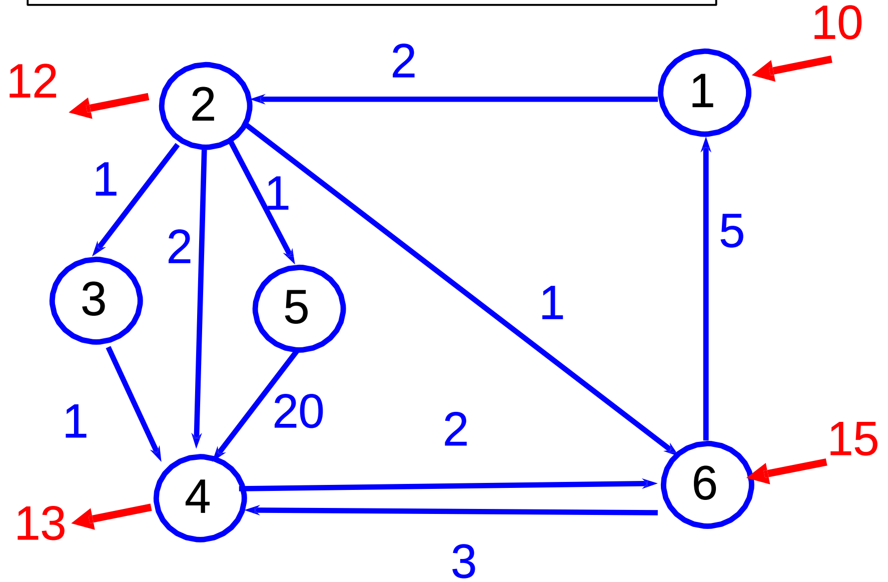
Reduced costs



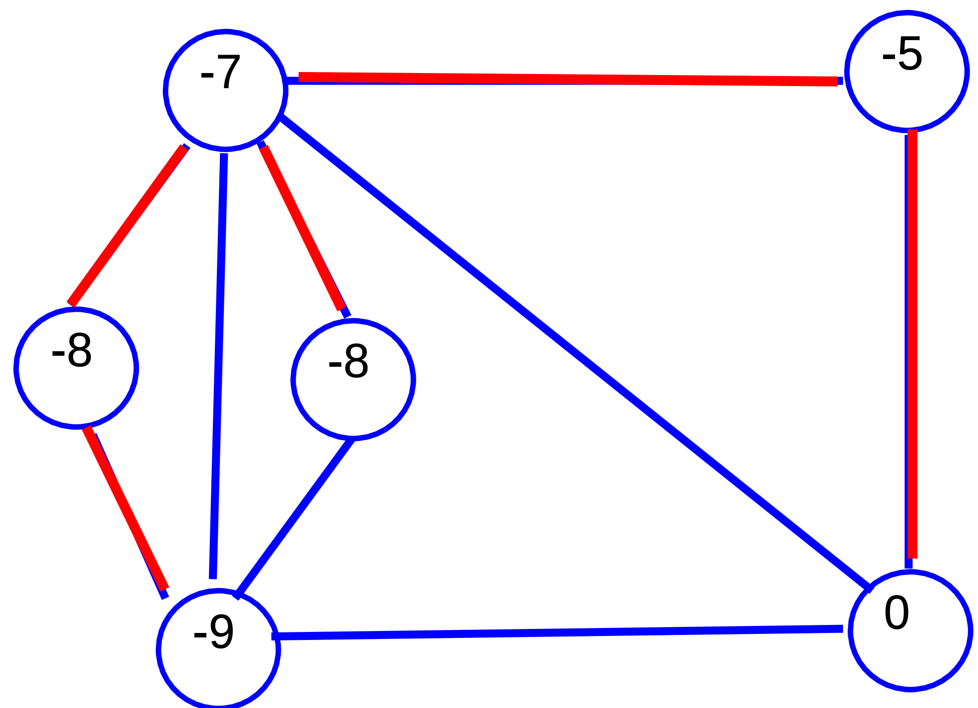
PROBLEMS with UPPER BOUNDS. $I_{N+} \leftrightarrow I_{N-}$

$$c^\top = (2, 1, 1, 2, 1, 20, 1, 5, 2, 3)$$

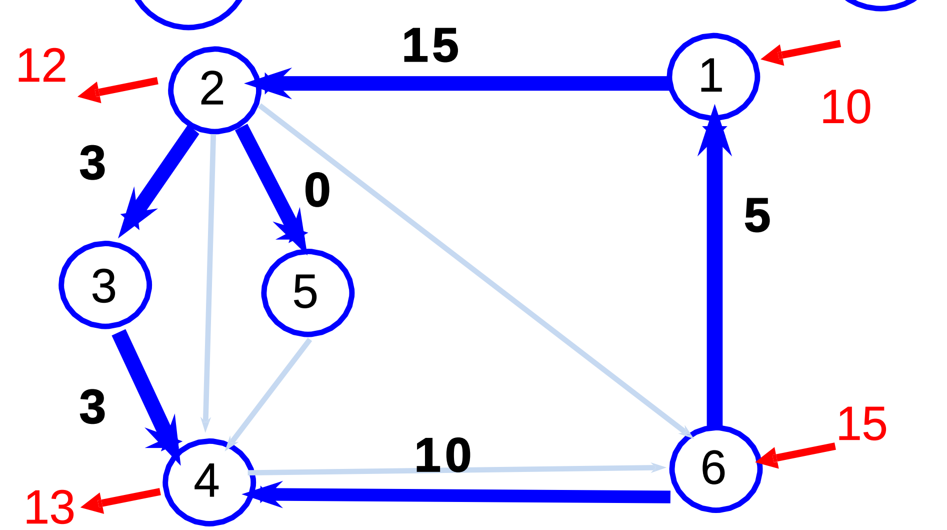
Link Costs



Dual variables using the Spanning Tree

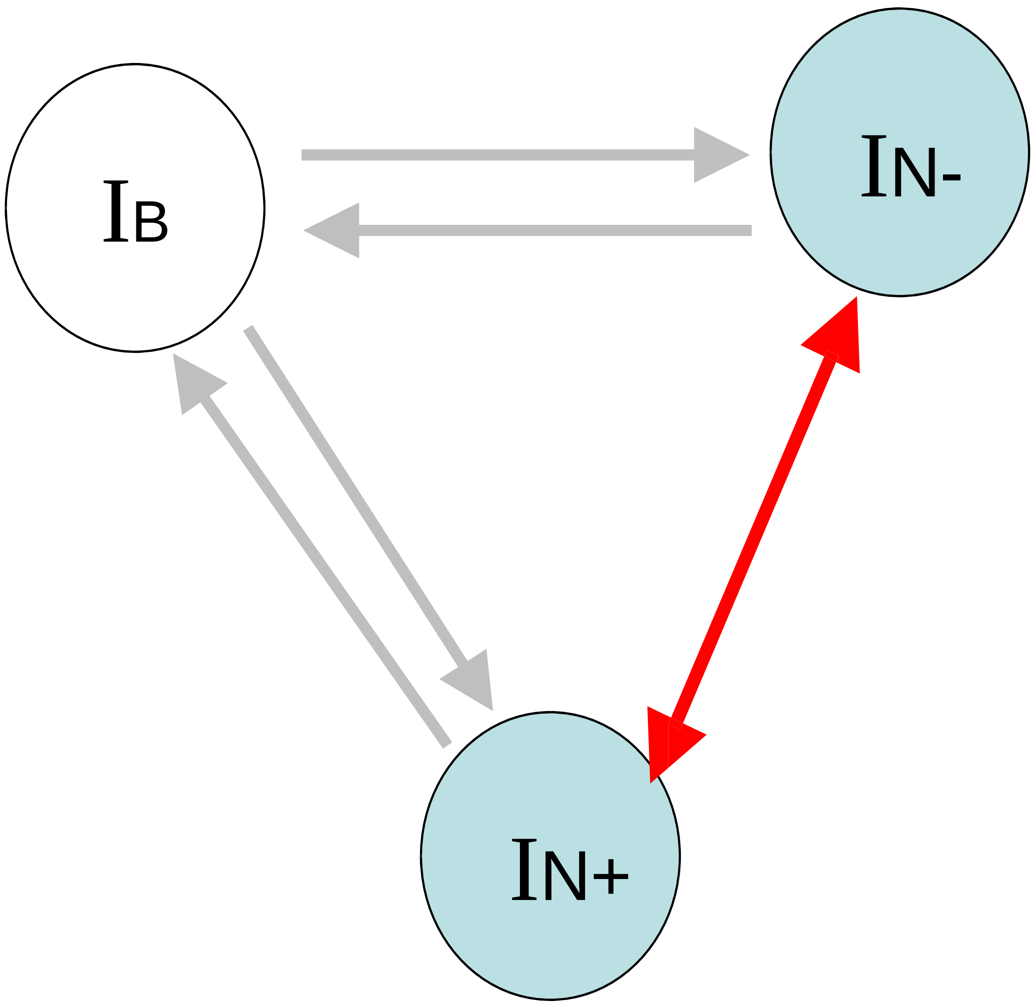


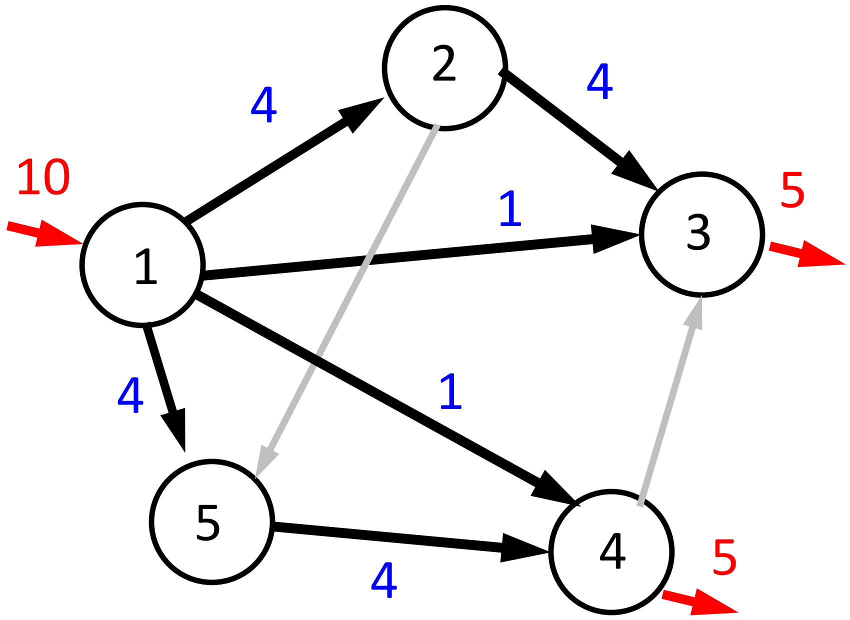
I_B={ (6,1), (1,2), (2,3), (3,4), (2,5) }
I_{N+}= { (6,4) }
I_{N-} = the remaining links



Check that this solution is optimal

- Interchange $I_{N+} \leftarrow\rightarrow I_{N-}$





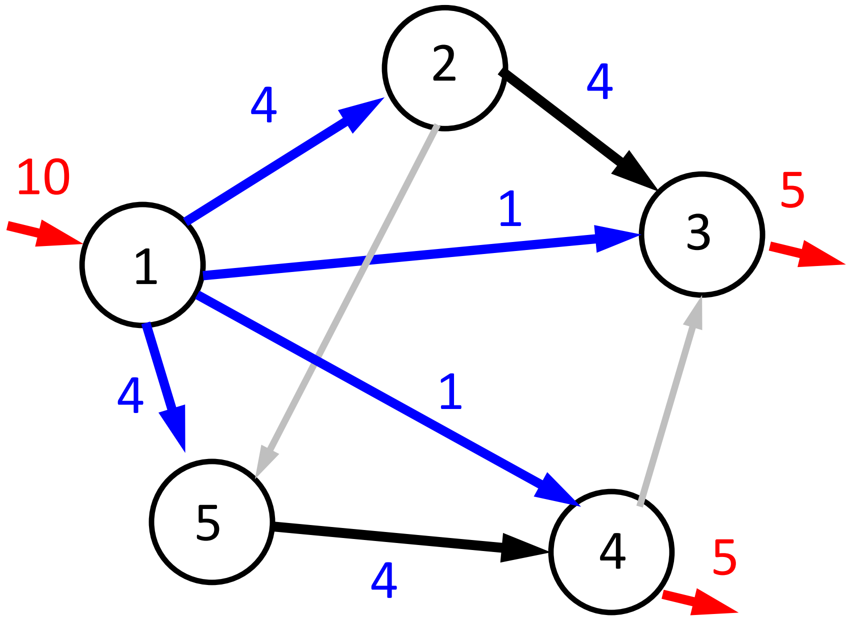
Assume the Min-Cost Flow problem defined by c_i=1, u_{ij}=4

Consider the initial feasible solution shown here. Try to find the optimal solution of the problema.

- State which are the set of basic índices I_B
- Id. Non-basic I_{N+} , I_{N-}

PROBLEMS with UPPER BOUNDS.

An example for the interchange $I_{N+} \leftarrow\rightarrow I_B$



Assume the Min-Cost Flow problem defined by c_i=1, u_{ij}=4

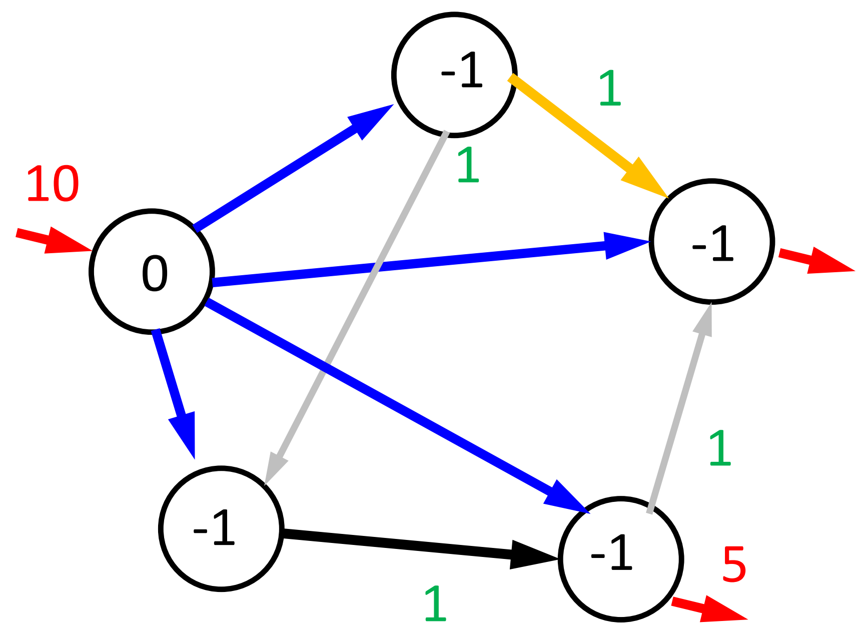
Consider the initial feasible solution shown here. Try to find the optimal solution of the problema.

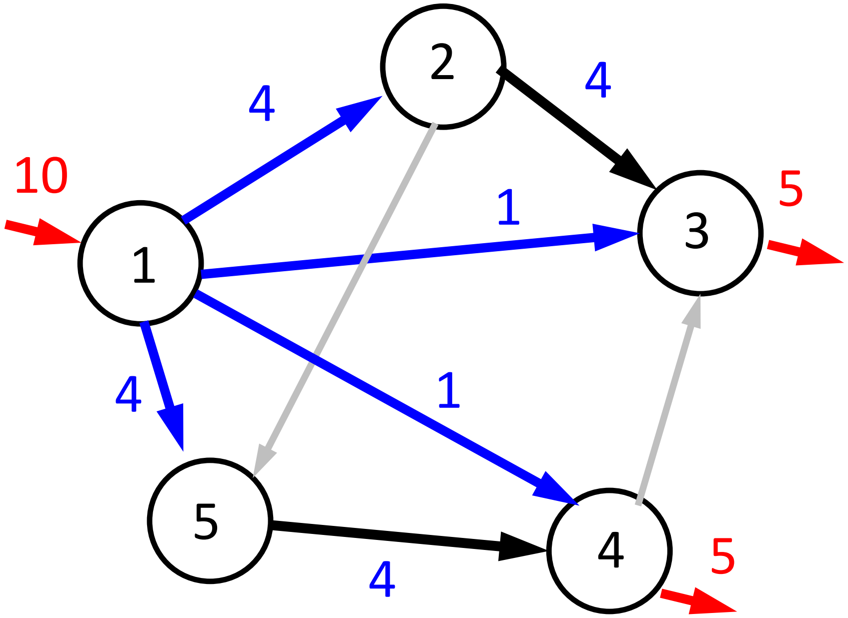
- State which are the set of basic índices I_B
- Id. Non-basic I_{N+} , I_{N-}

I_B ={(1,2), (1,3), (1,4), (1,5)}
I_{N+} ={(2,3), (5,4)}
I_{N-} ={(2,5), (4,3) }

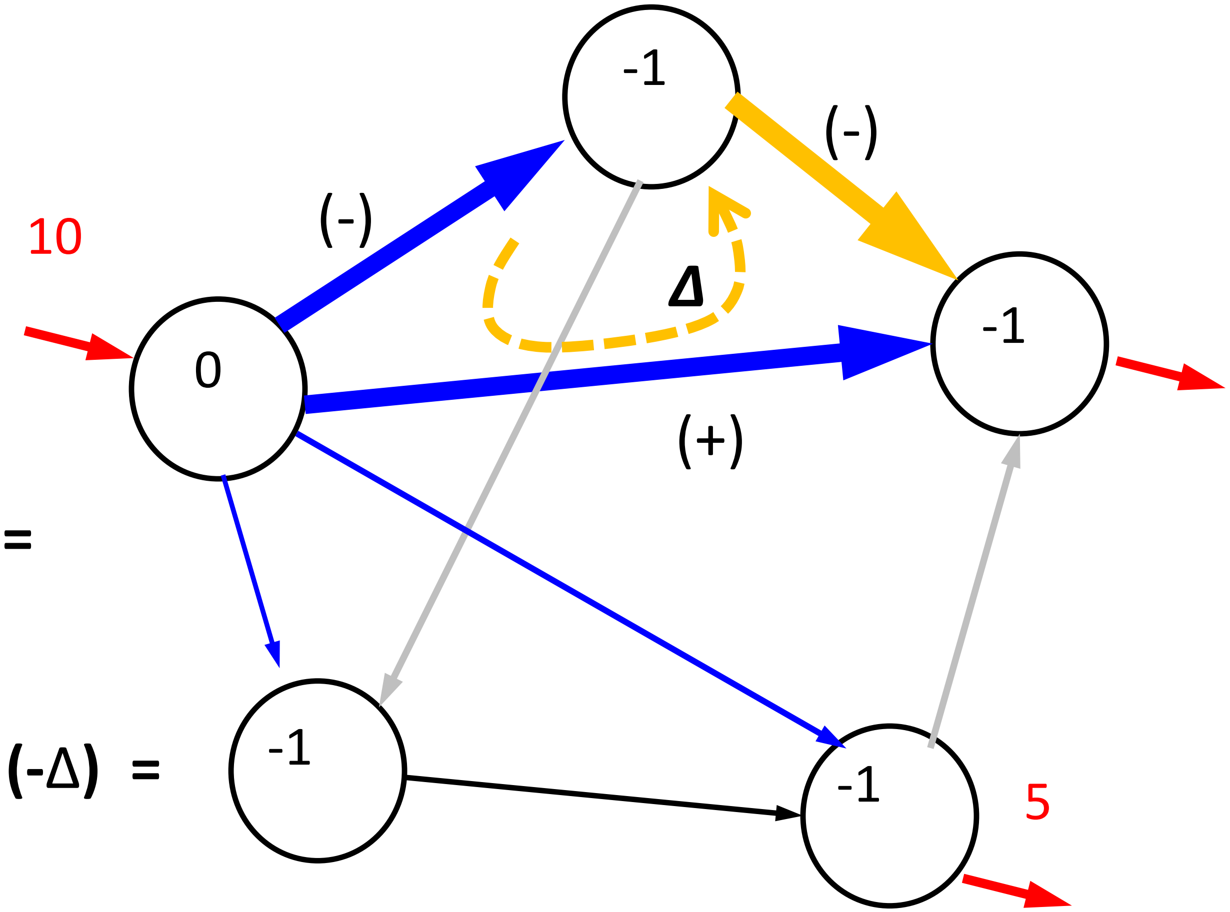
Current Obj.F. Value = 18

Either (2,3) or (5,4) may leave I_{N+} providing a better obj. Functon value.
- (2,3) is chosen





How much can be the recirculating flow Δ ??

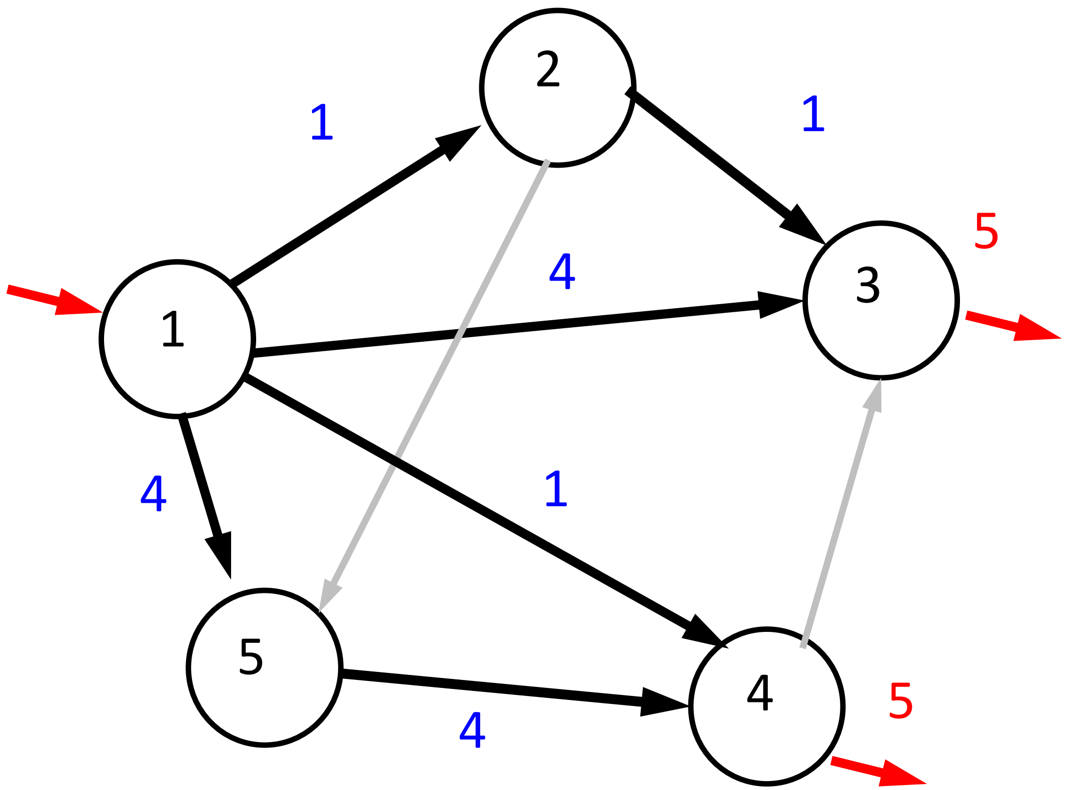


Δ = Min{ $u_{13} - x_{13}$, x_{23} , x_{12} **} =**
Min{ 3, 4, 4 **} = 3**

New Obj.F = Old Obj.F + r_{23} **(-Δ) =**
= 18 + 1 x (-3) = **15**

New feasible solution.

Can you identify the new index sets I_{B}, $I_{\mathrm{N+}}$, $I_{\mathrm{N+}}$???

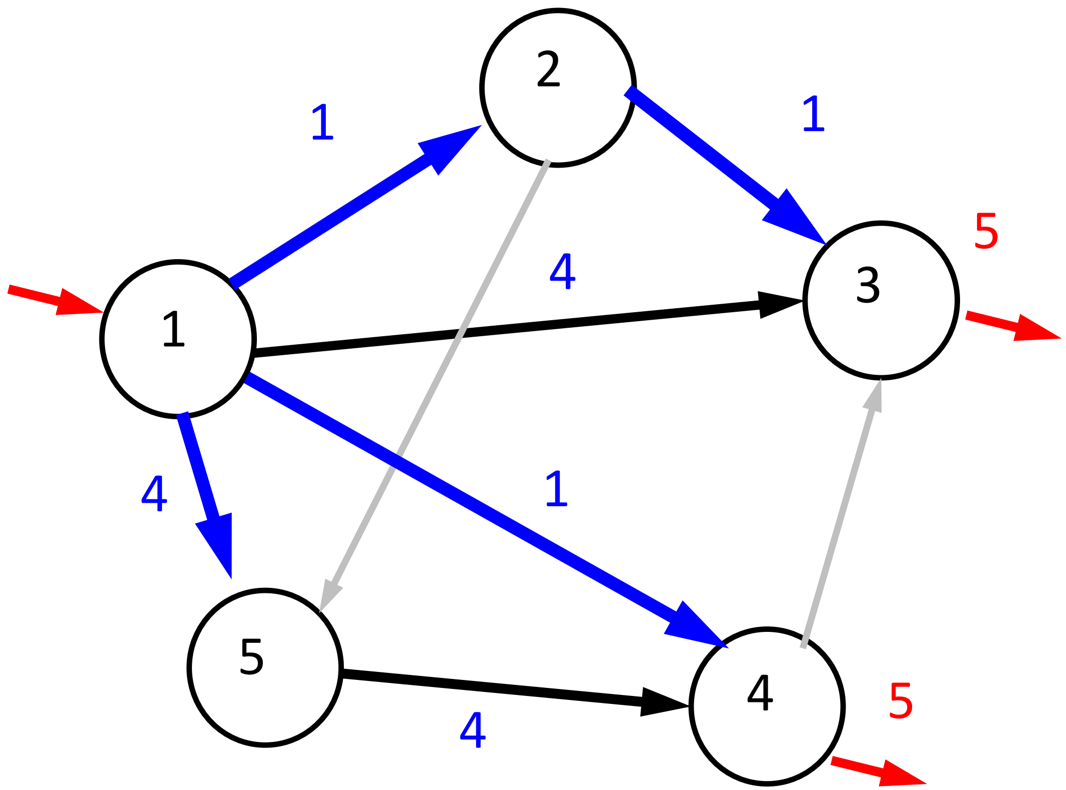


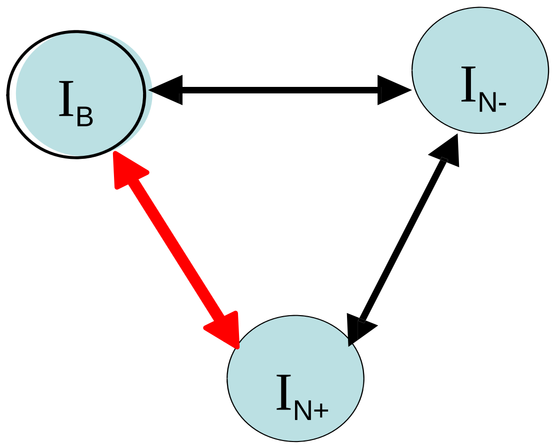
New feasible solution:

I_B ={(1,2), (2,3), (1,5), (1,4)},

I_{N+} ={(1,3), (5,4)} , I_{N-}= {(2,5), (4,5)}

Why???





$$I_{N^+} = \{ a \notin I_B, | x_a = u_a \}, \; I_{N^-} = \{ a \notin I_B, | x_a = 0 \}$$

`SIMPLEX Algorithm for Network Flow Problems (with upper bounds)`

0. Find an initial Feasible Basic Solution.
   Fix a node as the root node and set its dual variable to 0.

1. - Calculate dual variables $\lambda_i, i \in N$, for the current basis I_B
   (Remember: for basic links $a = (i, j) \in I_B$, $c_{ij} = \lambda_i - \lambda_j$)

2. - Evaluate reduced costs for non-basic links:
   $r_{i,j} = c_{i,j} - (\lambda_i - \lambda_j), (i, j) \in I_{N^+} \cup I_{N^-}$

3. **IF** (OPTIMALITY CRITERION) IS **NOT** MET

   a) Find some $a \in I_{N^+}$ with $r_a > 0$ **OR** some $a \in I_{N^-}$ with $r_a < 0$

   b) Consider such $a = (i, j)$; identify a cycle.

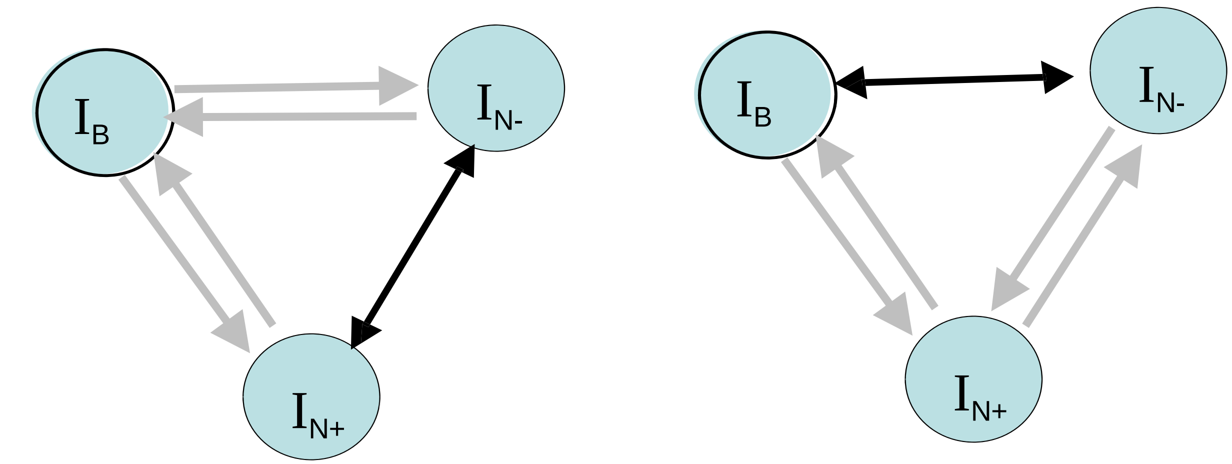
   c) Calculate the max. incr./decr. flow on that cycle $((i, j) \in I_{N-}/\ (i, j) \in I_{N+})$.

   d) Calculate new flows and index sets I_B, I_{N+}, I_{N-}

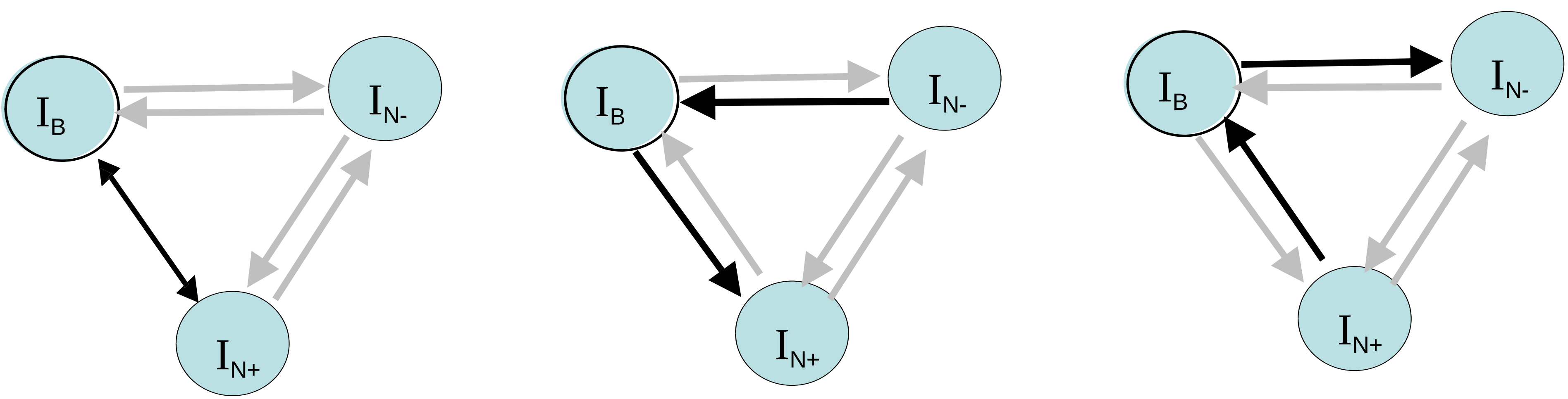
4. Go To 1

OPTIMALITY CRITERION =

$= r_a \leq 0, \forall a \in I_{N^+}$ **AND** $r_a \geq 0 \; a \in I_{N^-}, \forall a \in I_{N^-}$



Always $|I_B| = |N|-1$



- Step 3.d in the SIMPLEX with upper bounds:

Possible interchanges for a chosen link (*i*, *j*) found at step 3.a