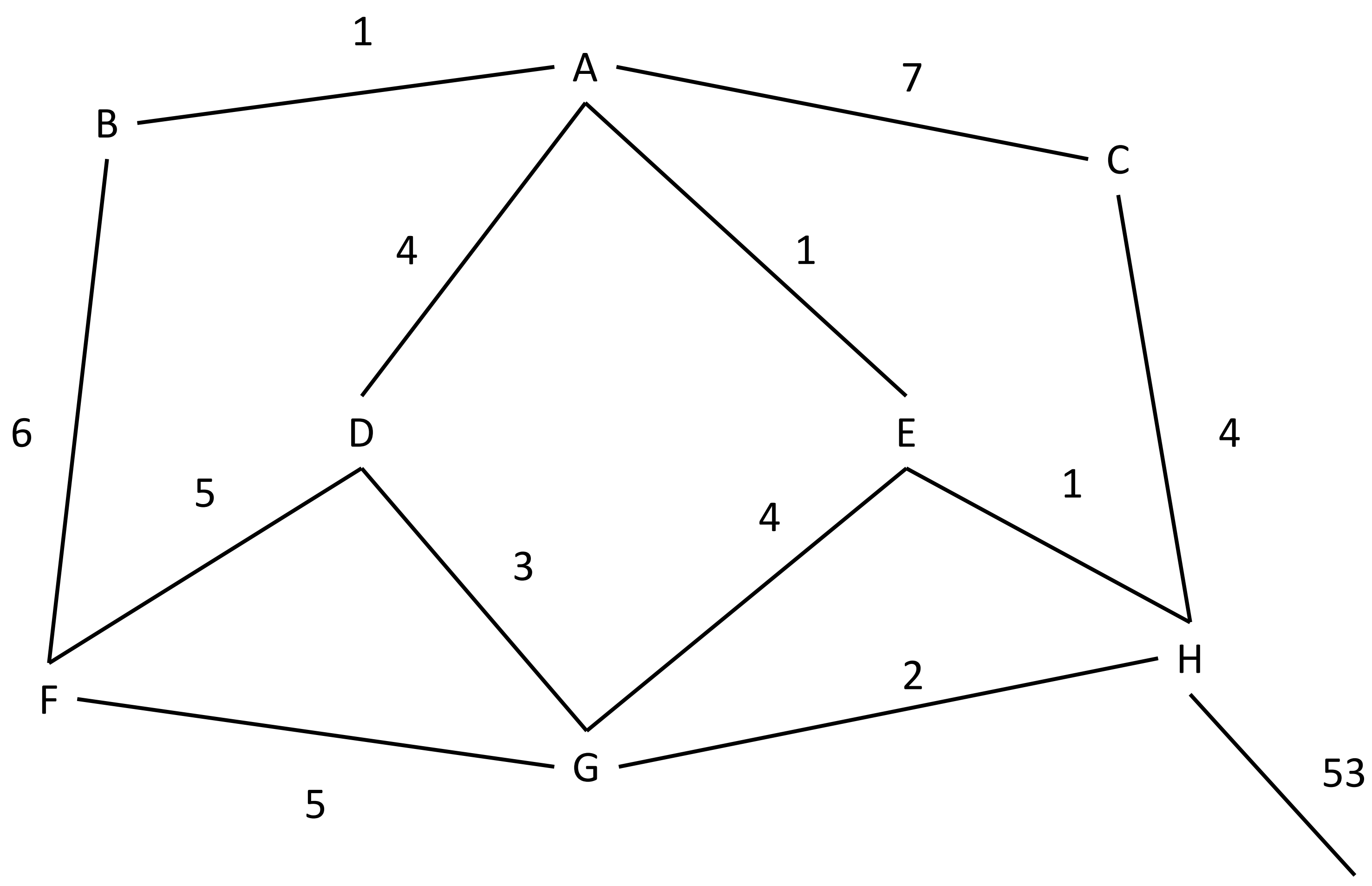
ALGORITHMES D’OPTIMISATION : ARBRE COUVRANT DE POIDS MINIMAL

Les données

- Graphes non orienté dont les arêtes ont un poids

Exemple



Représentation par matrice

	A	B	C	D	E	F	G	H	I
A	∞	1	7	4	1	∞	∞	∞	∞
B	1	∞	∞	∞	∞	6	∞	∞	∞
C	7	∞	∞	∞	∞	∞	∞	4	∞
D	4	∞	∞	∞	∞	5	3	∞	∞
E	1	∞	∞	∞	∞	∞	4	1	∞
F	∞	6	∞	5	∞	∞	5	∞	∞
G	∞	∞	∞	3	4	5	∞	2	∞
H	∞	∞	4	∞	1	∞	2	∞	53
I	∞	∞	∞	∞	∞	∞	∞	53	∞

Représentation Par listes d’adjacence

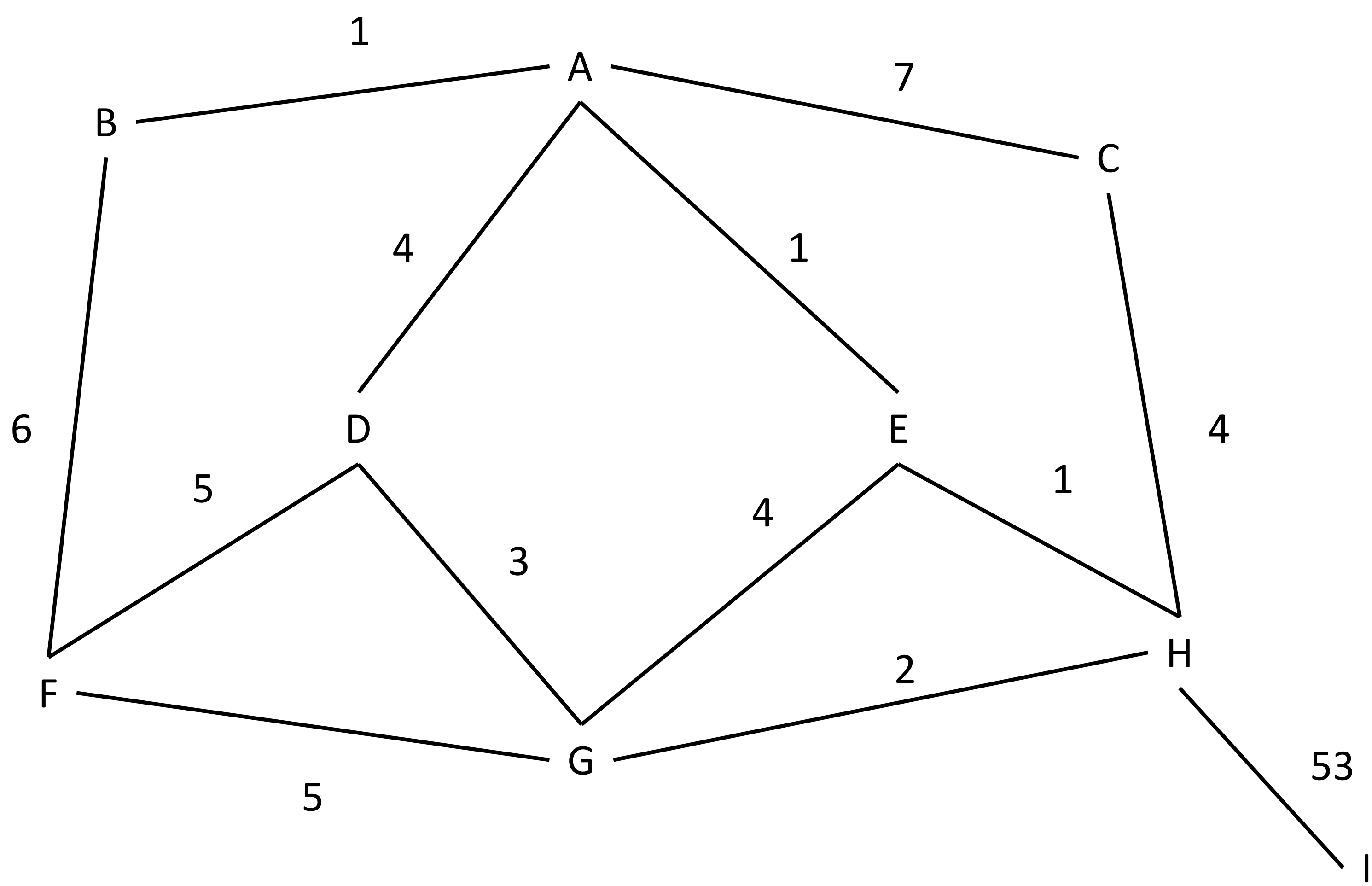
A	**<(B, 1), (C, 7), (D, 4), (E, 1)>**
B	<(A, 1), (F, 6)>
C	<(A, 7), (H, 4)>
D	<(A, 4), (F, 5), (G, 3)>
E	<(A, 1), (G, 4), (H, 1)>
F	<(B, 6), (D, 5), (G, 5)>
G	<(D, 3), (E, 4), (F, 5), (H, 2)>
H	<(C, 4), (E, 1), (G, 2), (I, 53,)>
I	<(H, 53)>

PARTIE 1 : PRÉSENTATION DU PROBLÈME

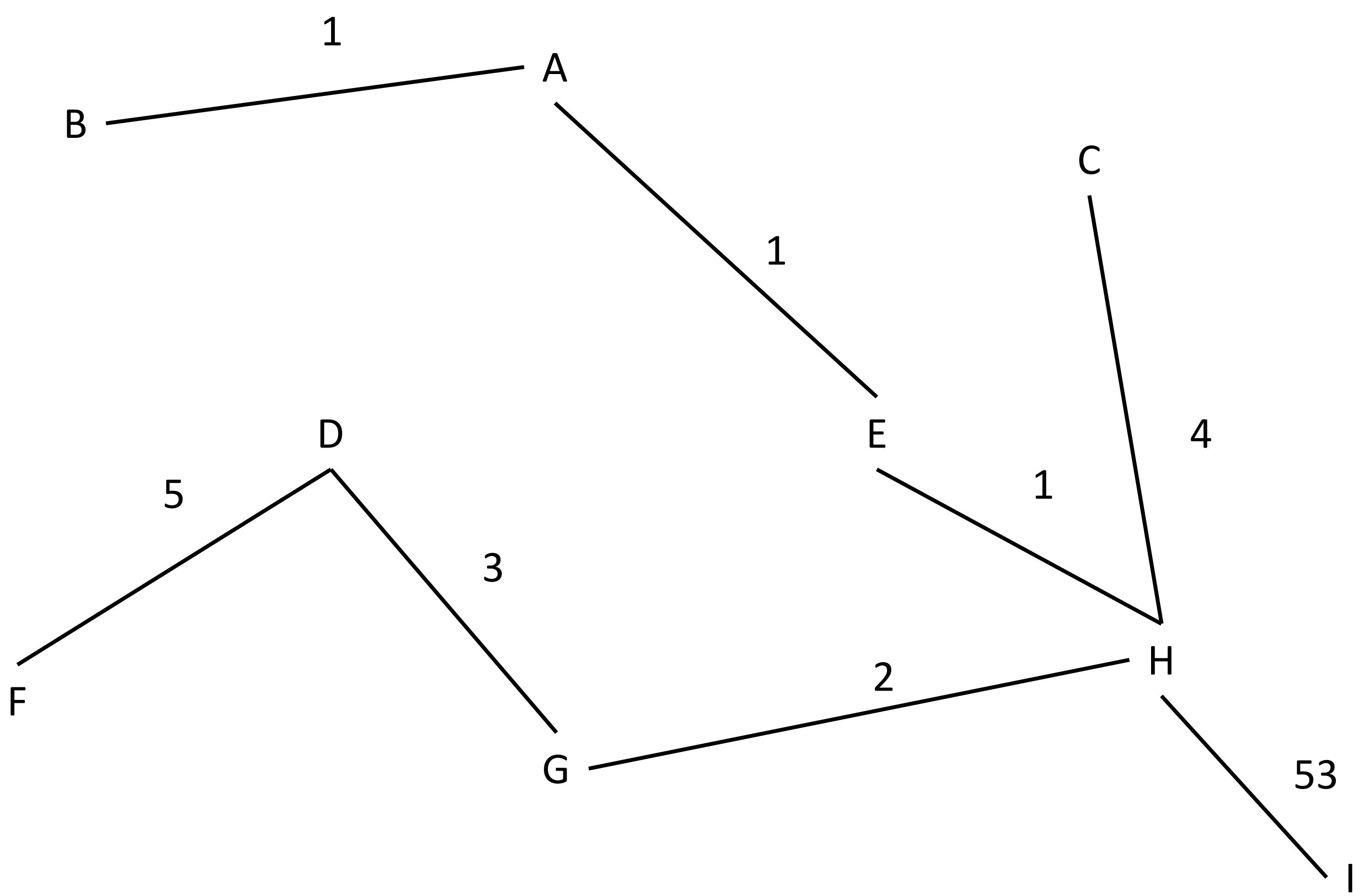
Définition du problème

- Etant donné un graphe non orienté connexe dont les arêtes sont pondérées G = (X, U, V).
- On veut calculer G’ = (X, U’) tel que
  1. G’ est un graphe partiel de G
  2. G’ est un Graphe Connexe sans cycles (Un arbre)
  3. Pour tout graphe partiel connexe G’’ = (X,U’’), $\sum_{xy \in U'} V(xy) \leq \sum_{ab \in U''} V(ab)$
     i.e. c’est une structure de poids Minimal

Exemple



Exemple



Propriétés

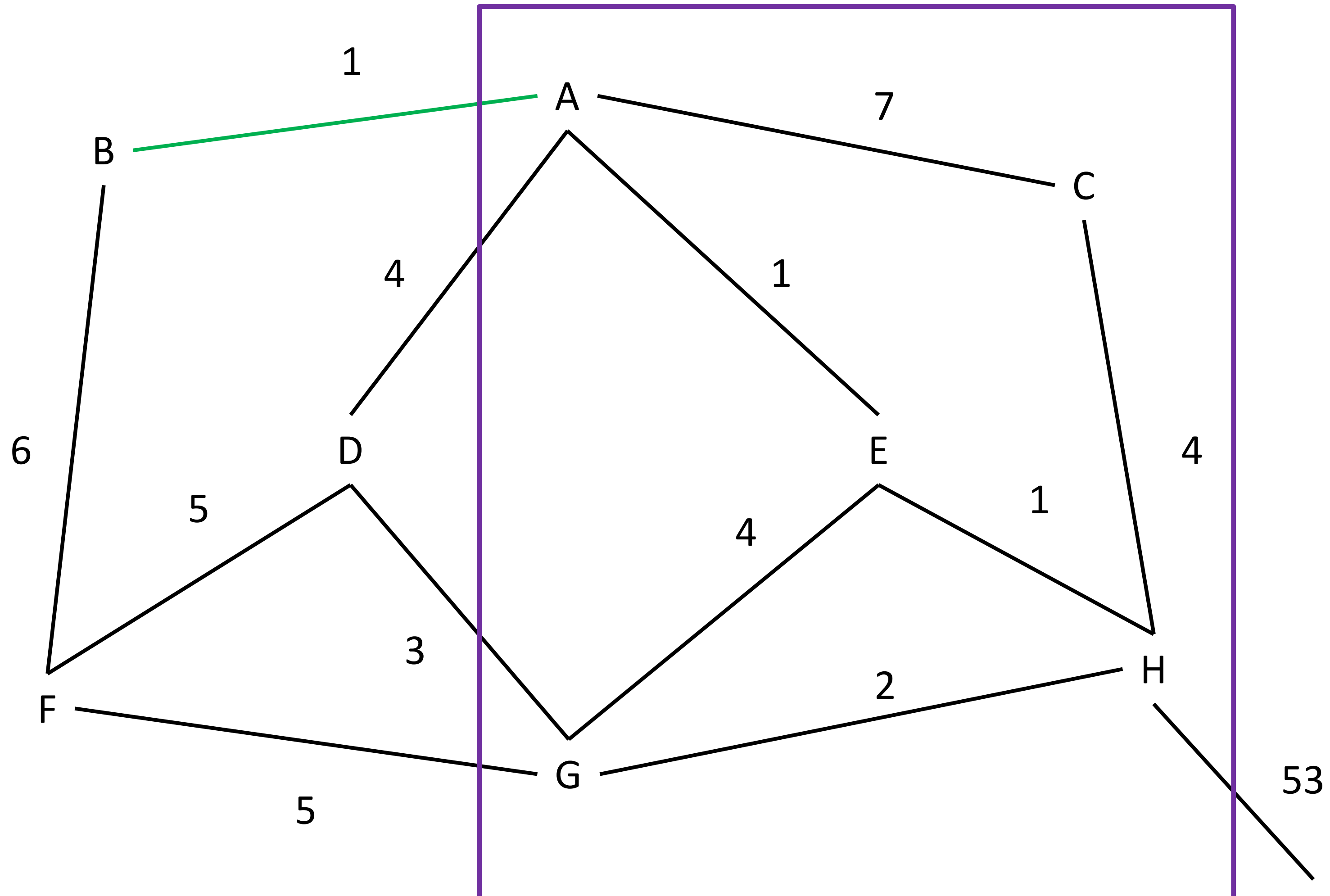
- Lorsque chaque poids d'arête est strictement positif, le sous graphe partiel connexe de poids minimal est un arbre couvrant.
  - Idée de Preuve si ce graphe partiel n'est pas un arbre alors il contient un cycle élémentaire contenant au moins trois sommets et donc au moins trois arêtes. Le retrait de n'importe laquelle de ces arêtes maintient une structure couvrante de poids strictement inférieur (voir principe 2).

Principes des algorithmes (1)

- Soit X1, X2 une partition quelconque de X
- Soit A1 l'ensemble des arêtes ayant une extrémité dans X1 et une extrémité dans X2
- Soit A2 les arêtes de A1 de poids minimal.

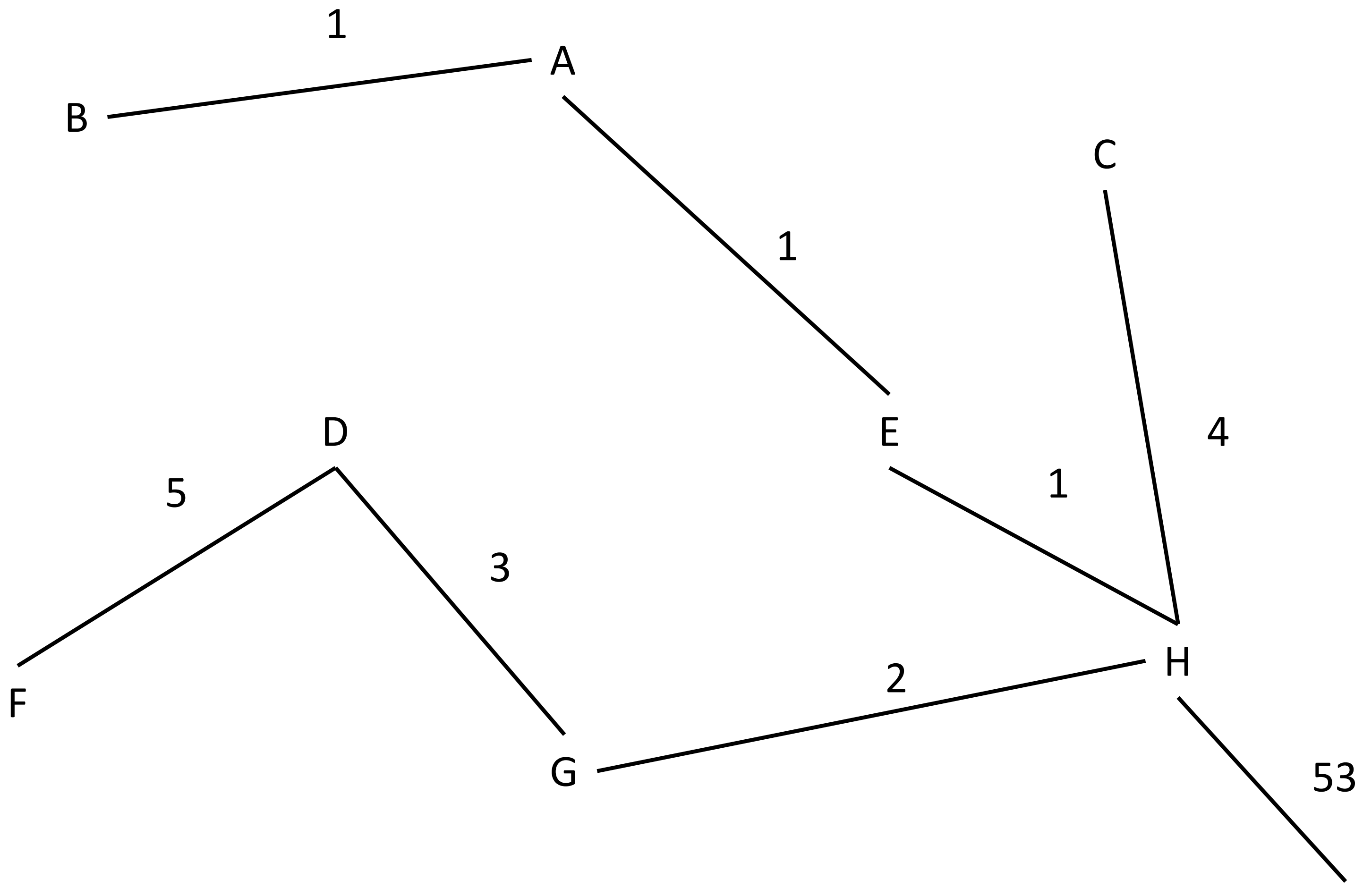
- Tout arbre couvrant de poids minimal utilise au moins une arête de A2

Exemple

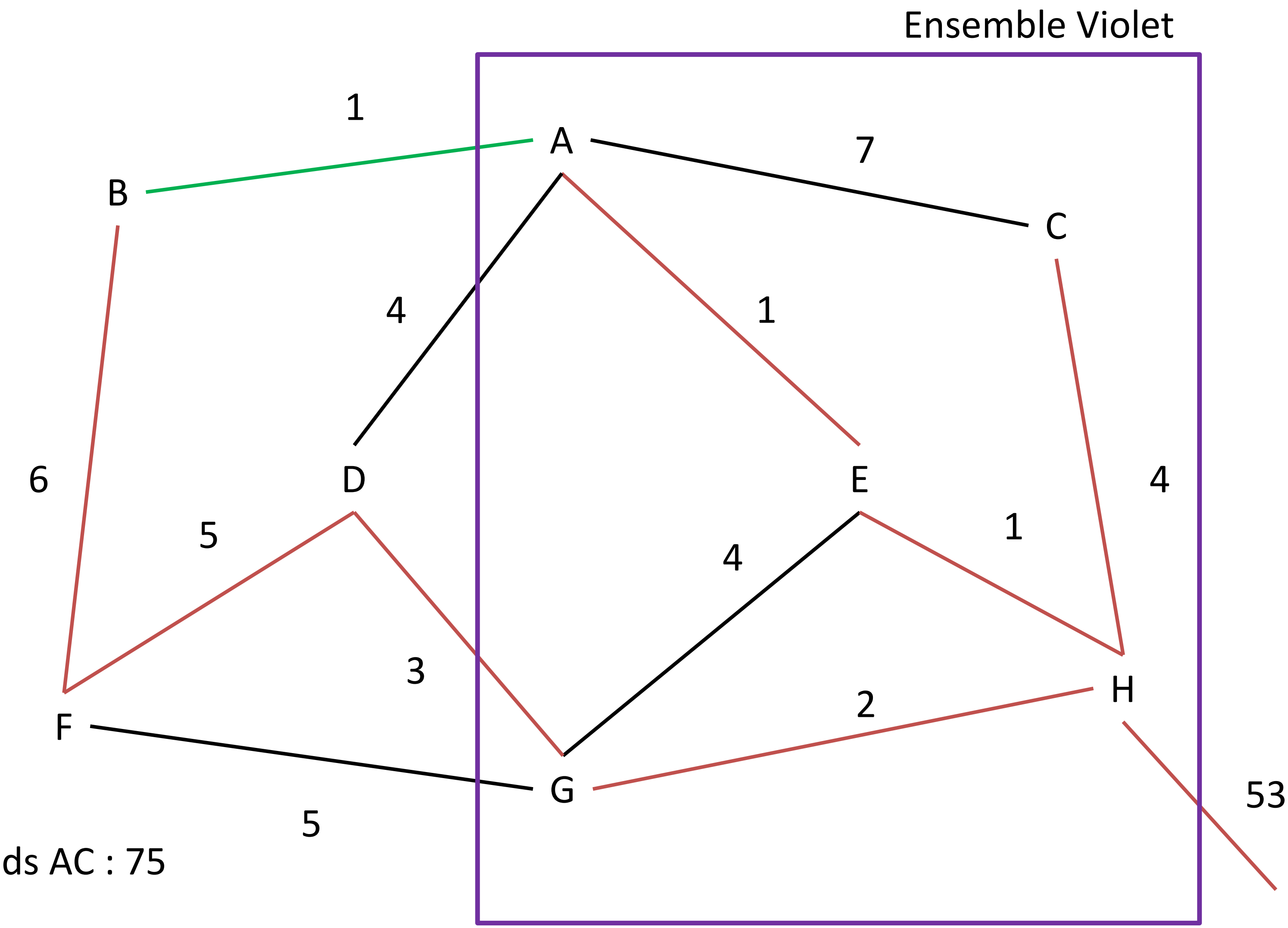


Exemple

L’arête AB
est présente

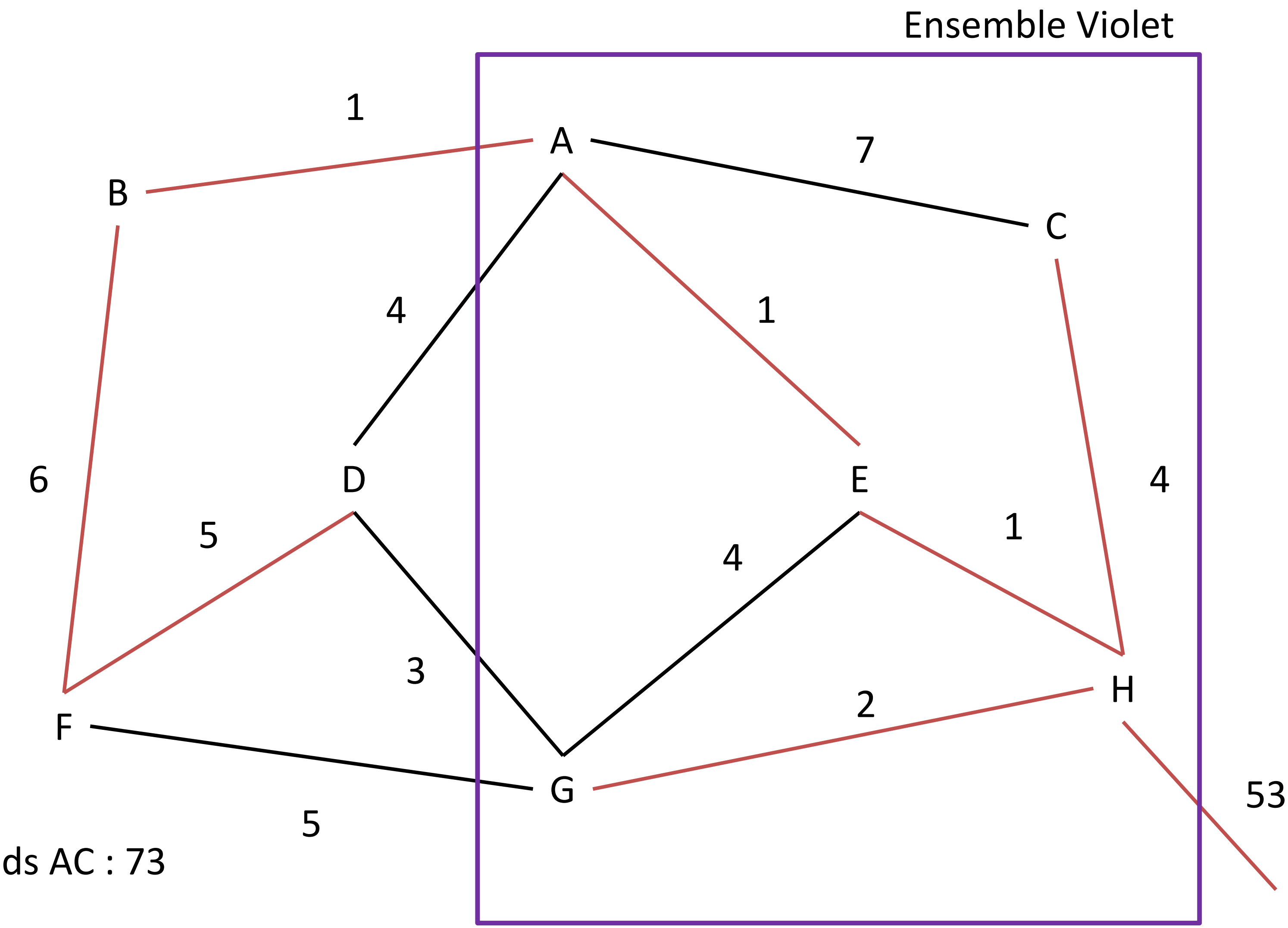


Preuve par contradiction



Poids AC : 75

Preuve



Poids AC : 73

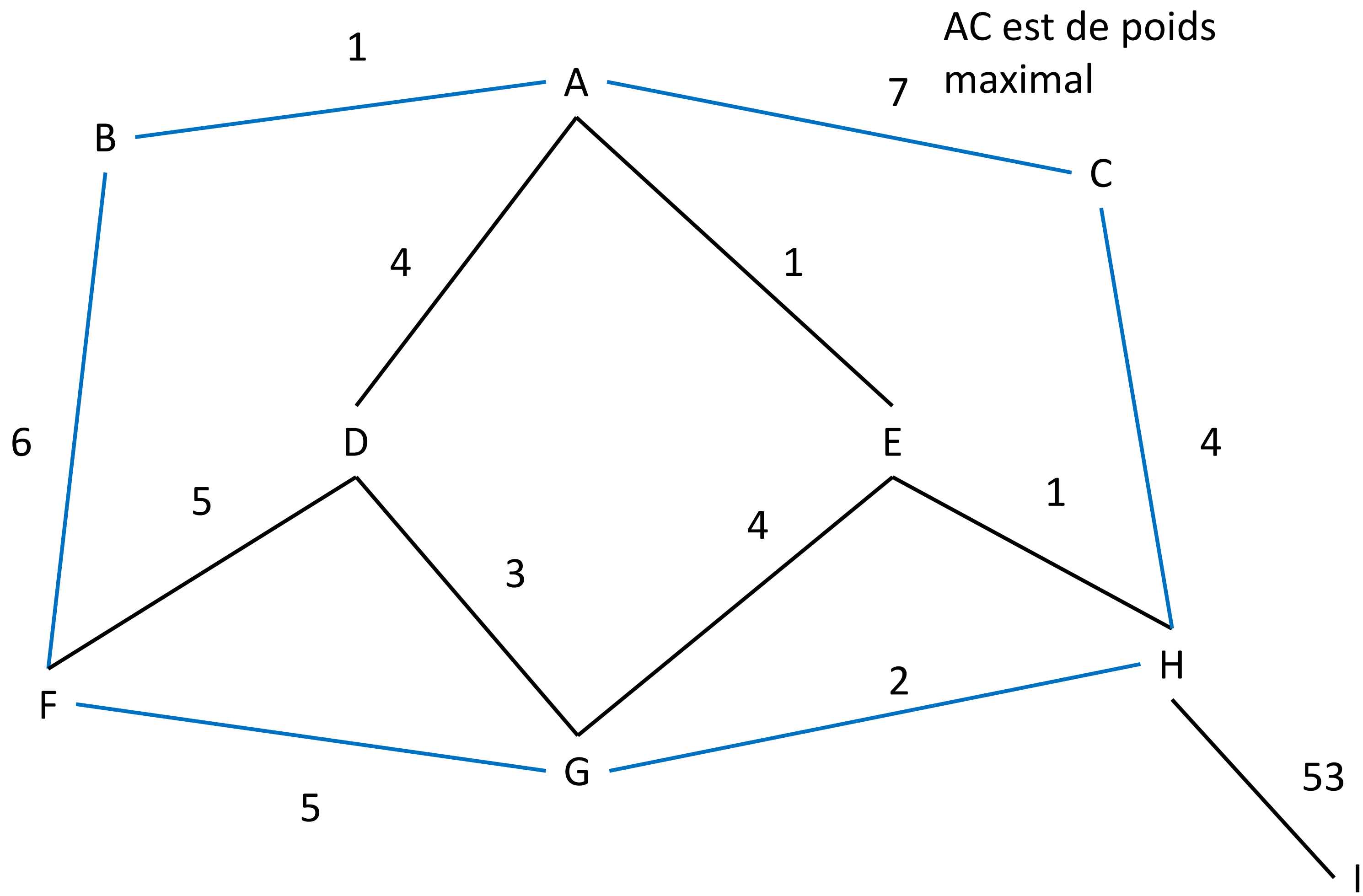
Principe des algorithmes (2)

- Pour tout cycle élémentaire C et pour tout arbre de poids minimal A

- Il existe une arête de poids maximal de C qui n’est pas présente dans A.

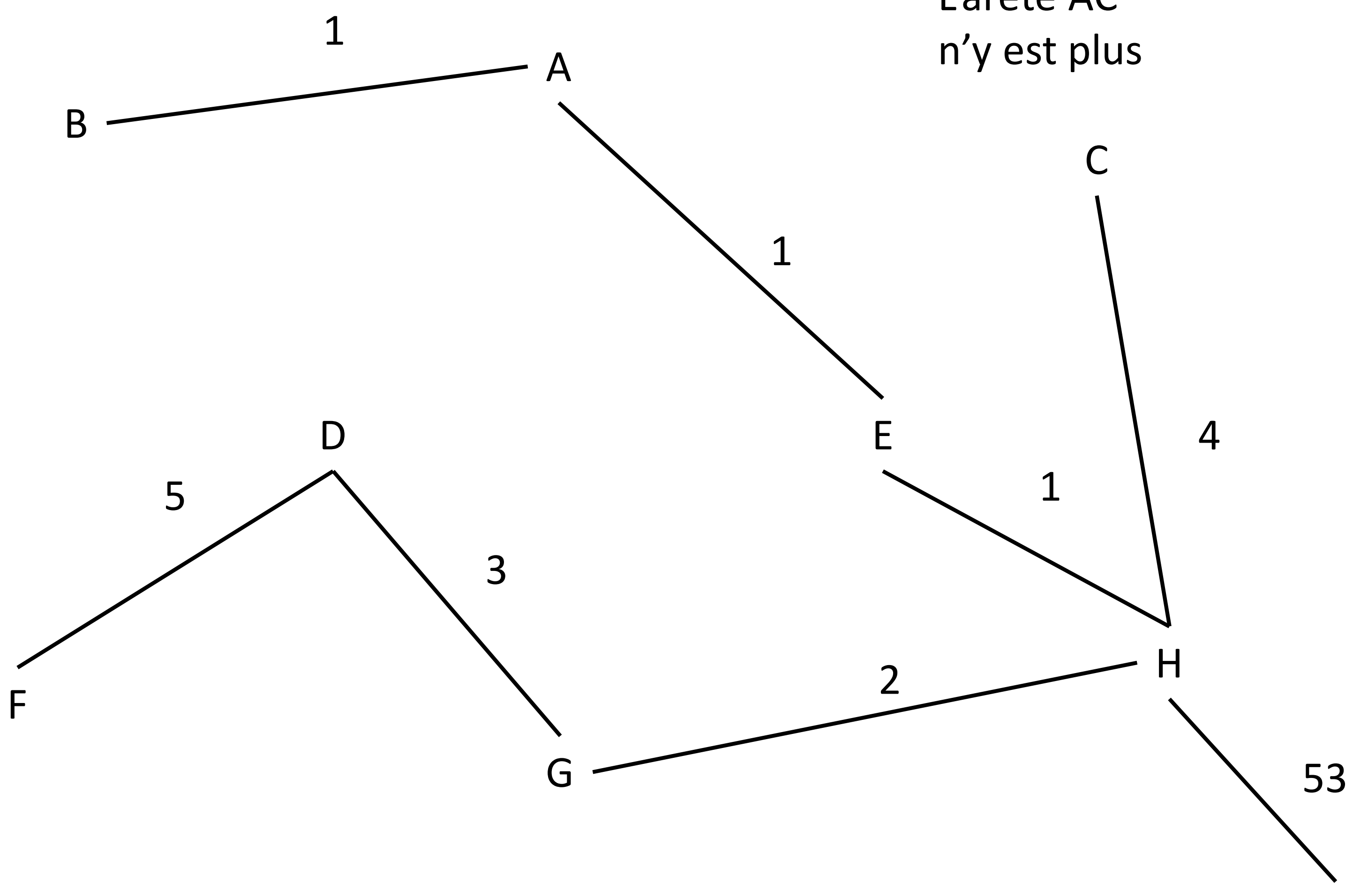
- Pour être plus précis on peut retirer n’importe quelle arête de poids maximal de C.

Exemple

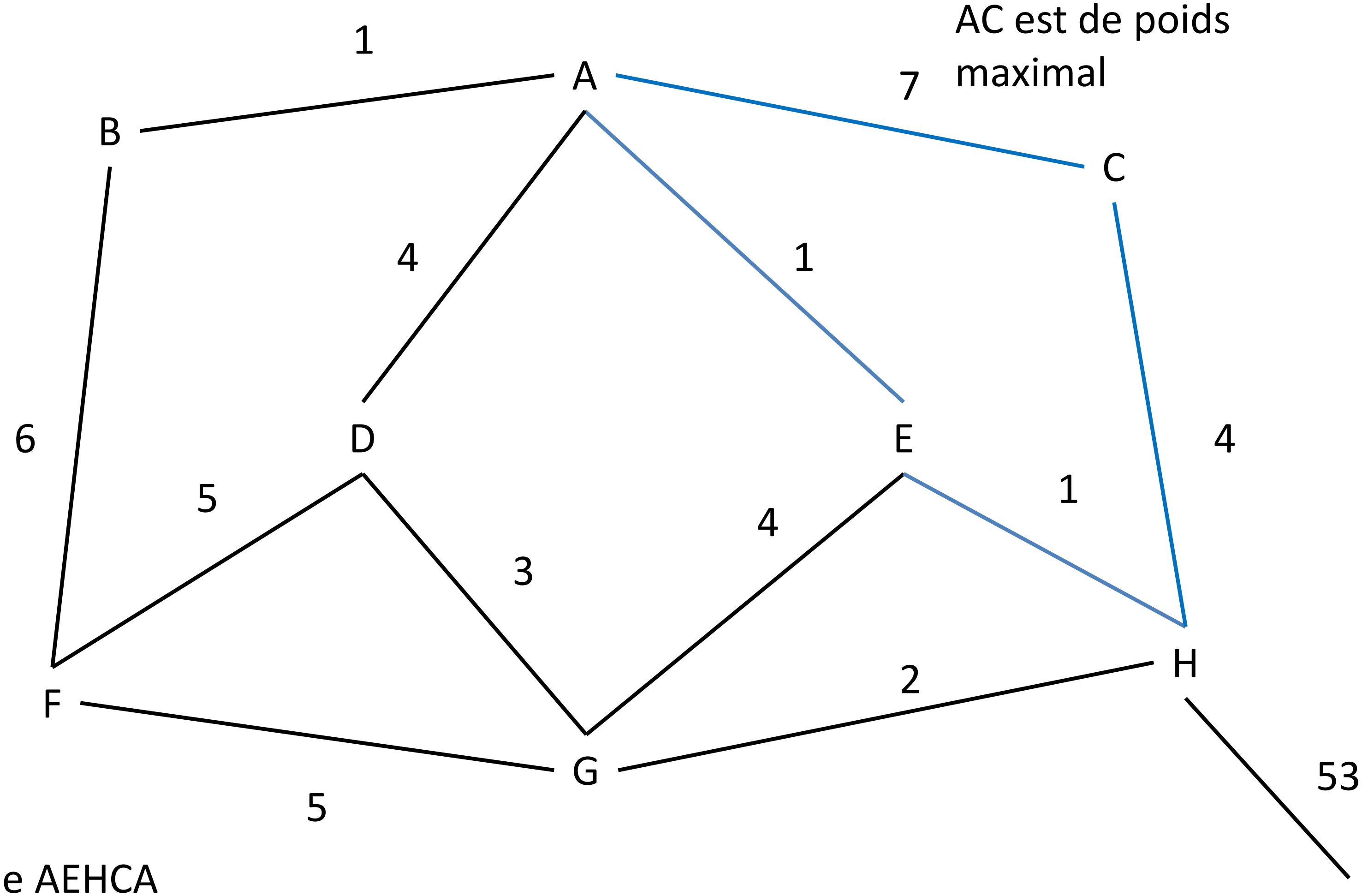


Exemple

L'arête AC
n'y est plus



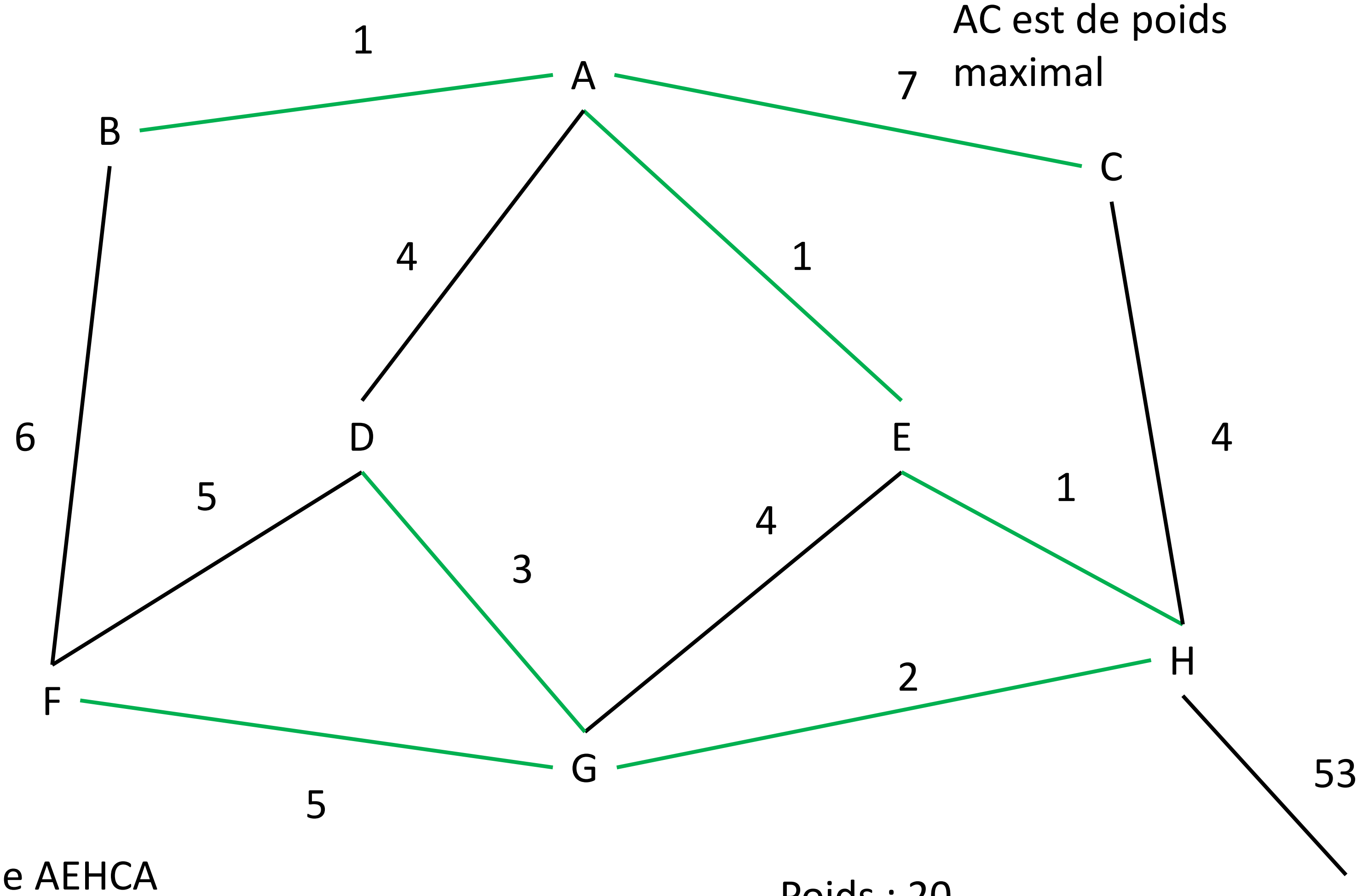
Preuve par contradiction



AC est de poids maximal

Cycle AEHCA

Preuve par contradiction

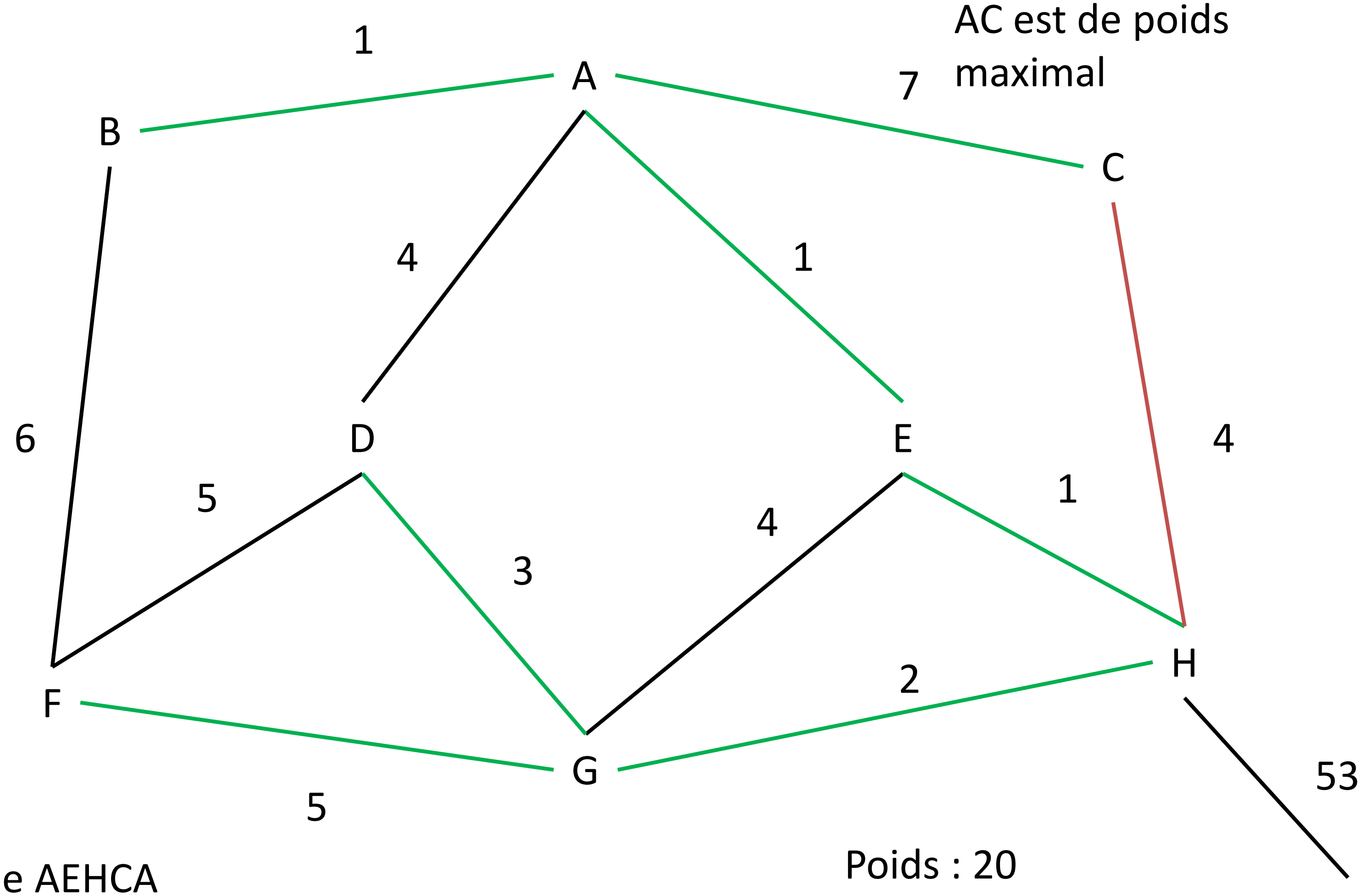


AC est de poids maximal

Cycle AEHCA

Poids : 20

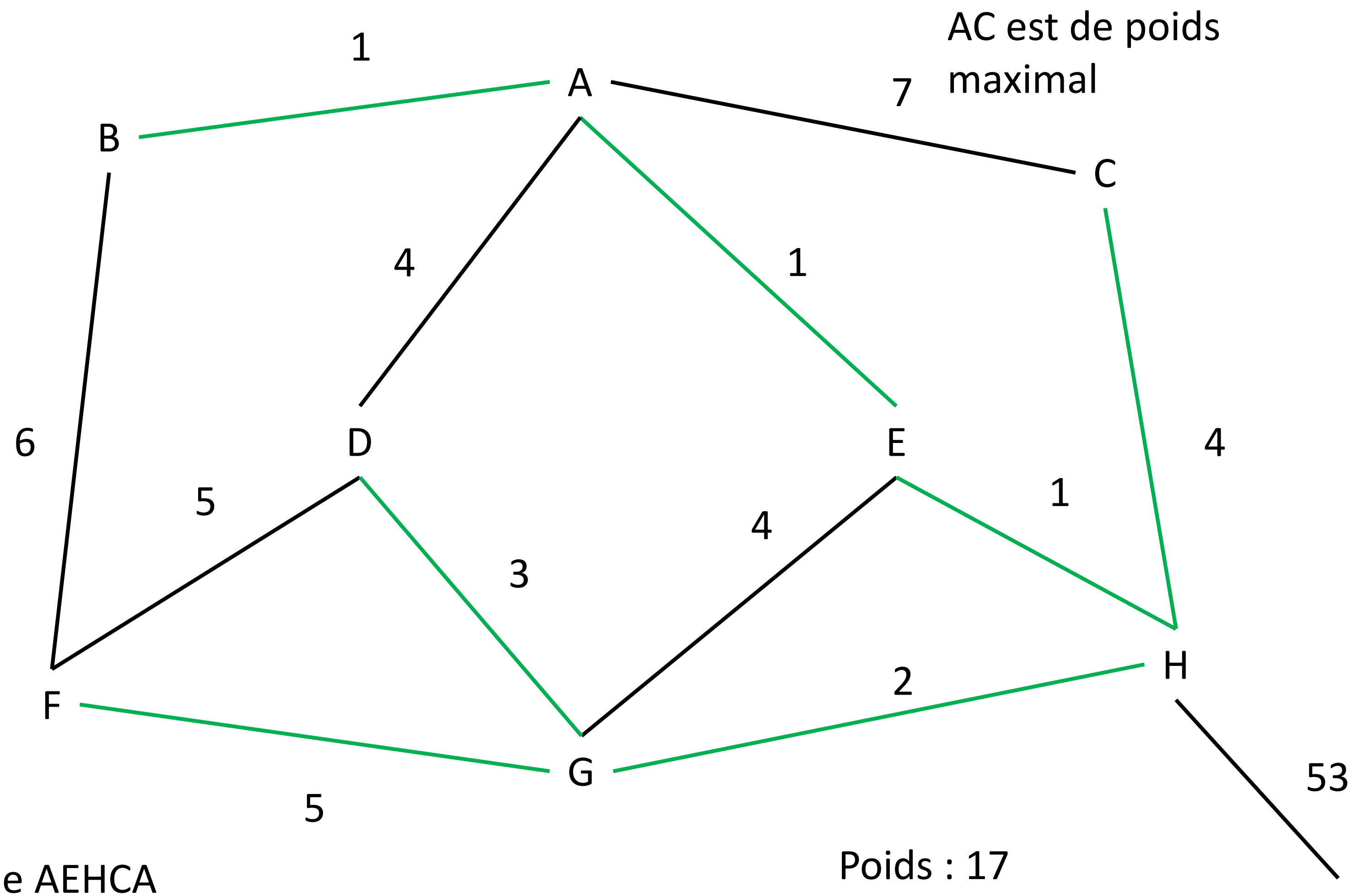
Preuve par contradiction



Cycle AEHCA

Poids : 20

Preuve par contradiction



Cycle AEHCA

Poids : 17

Algorithmes

- Ces deux principes permettent d'écrire des algorithmes de construction d'arbres couvrants de poids minimal dont les trois plus connus sont ceux de :
  - Prim (1959) ou Jarnik (1930)
  - Kruskal (1956)
  - Borůvka (1926) ou Choquet (1938)