

Exploration de graphes avec peu de mémoire

David ILCINKAS, LRI Orsay

En collaboration avec Pierre FRAIGNIAUD

Exploration de graphes

- But
 - l'entité mobile doit traverser au moins une fois chaque arête du graphe inconnu en un temps fini

Exploration de graphes

- **But**
 - l'entité mobile doit traverser au moins une fois chaque arête du graphe inconnu en un temps fini
- **Motivation**
 - exploration ou cartographie d'endroits inaccessibles à l'homme
 - maintenance d'un réseau informatique
 - recherche d'informations (P2P, Web)

Exploration de graphes

- But
 - l'entité mobile doit traverser au moins une fois chaque arête du graphe inconnu en un temps fini
- Motivation
 - exploration ou cartographie d'endroits inaccessibles à l'homme
 - maintenance d'un réseau informatique
 - recherche d'informations (P2P, Web)
- Hypothèse
 - graphes connexes
 - anonyme : manque d'information, senseurs limités

Graphes anonymes

Graphes anonymes

- Taille et topologie inconnues

Graphes anonymes

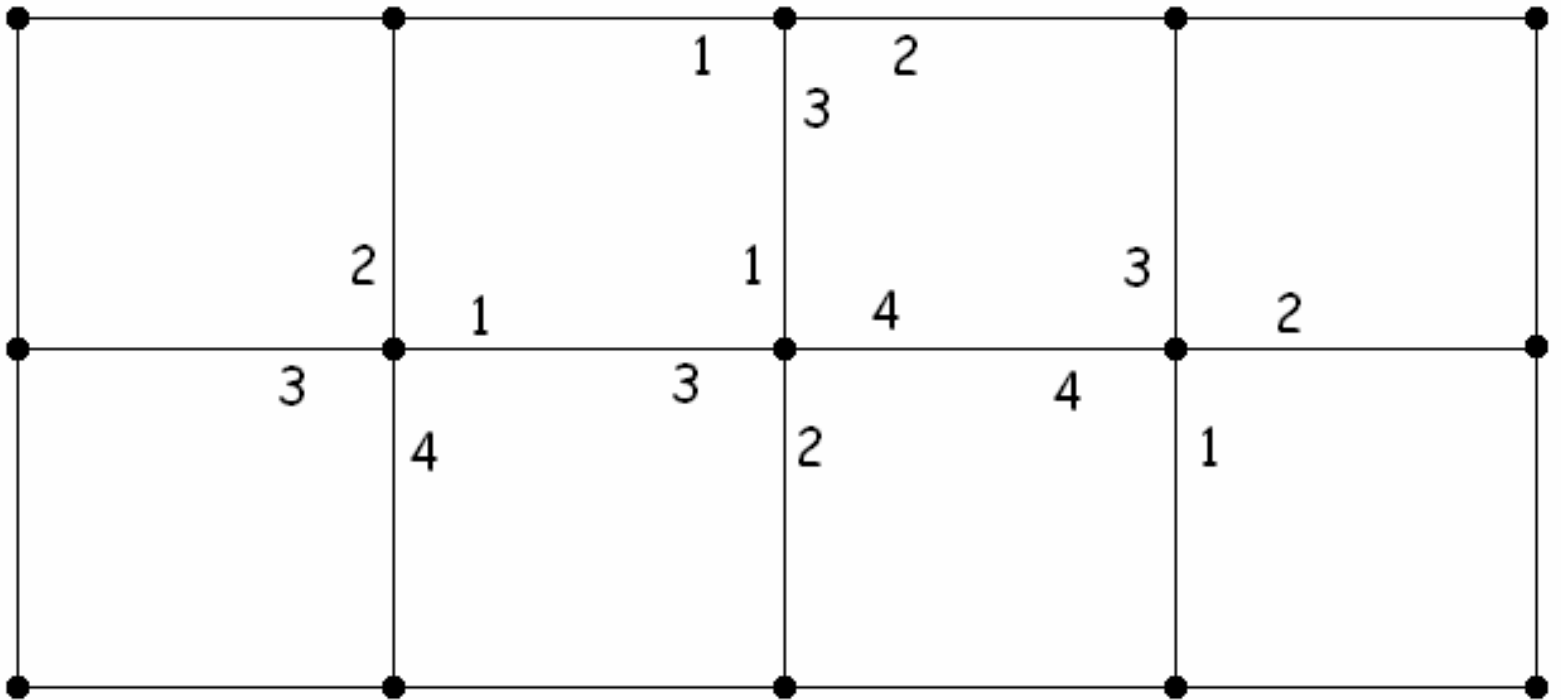
- Taille et topologie inconnues
- Pas d'étiquetage des sommets

Graphes anonymes

- Taille et topologie inconnues
- Pas d'étiquetage des sommets
- Numérotation locale des arêtes

Graphes anonymes

- Taille et topologie inconnues
- Pas d'étiquetage des sommets
- Numérotation locale des arêtes



L'agent mobile

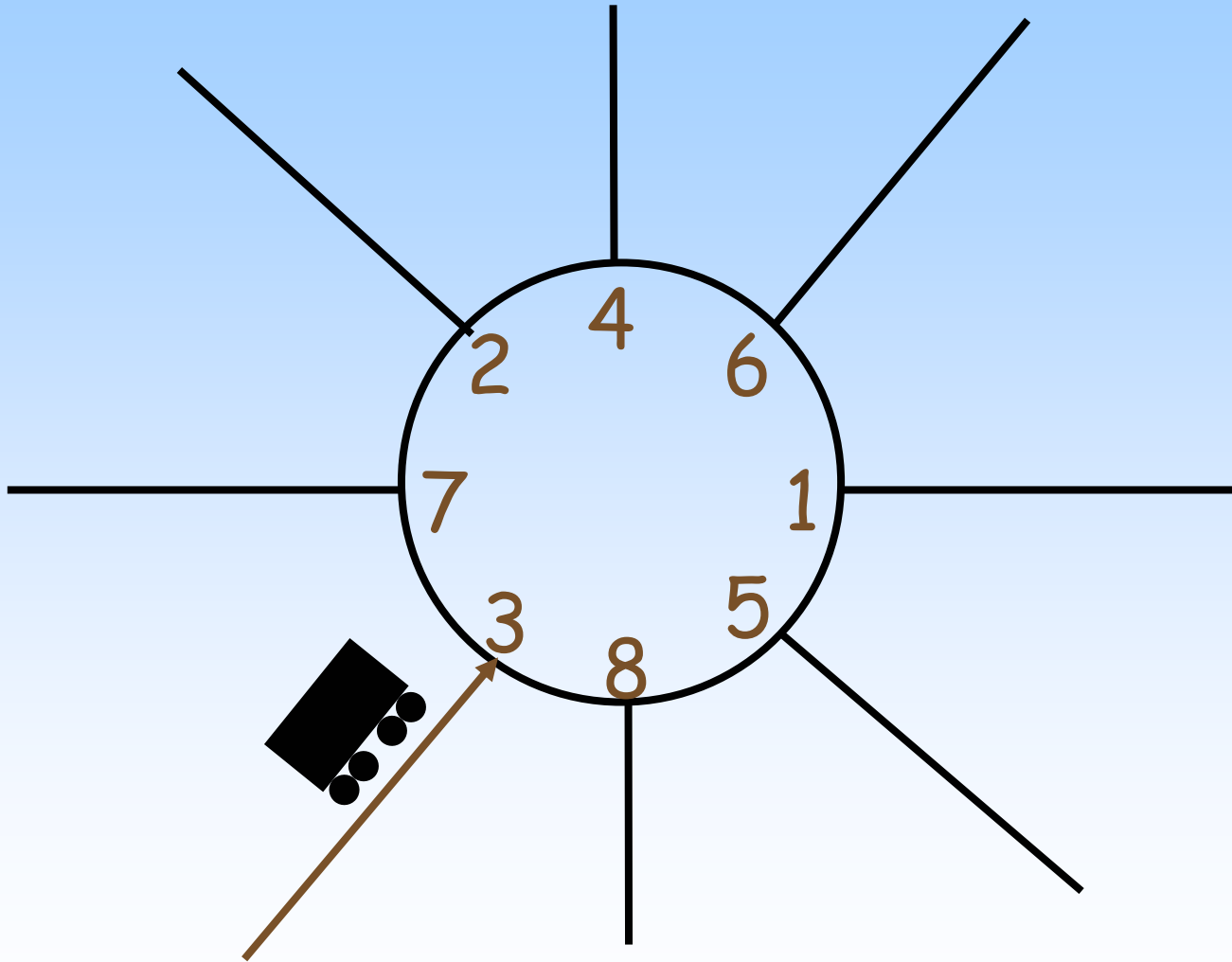
L'agent mobile

- **Automate** déterministe à **K états**
→ robot avec $\leq \log(K)$ bits de mémoire

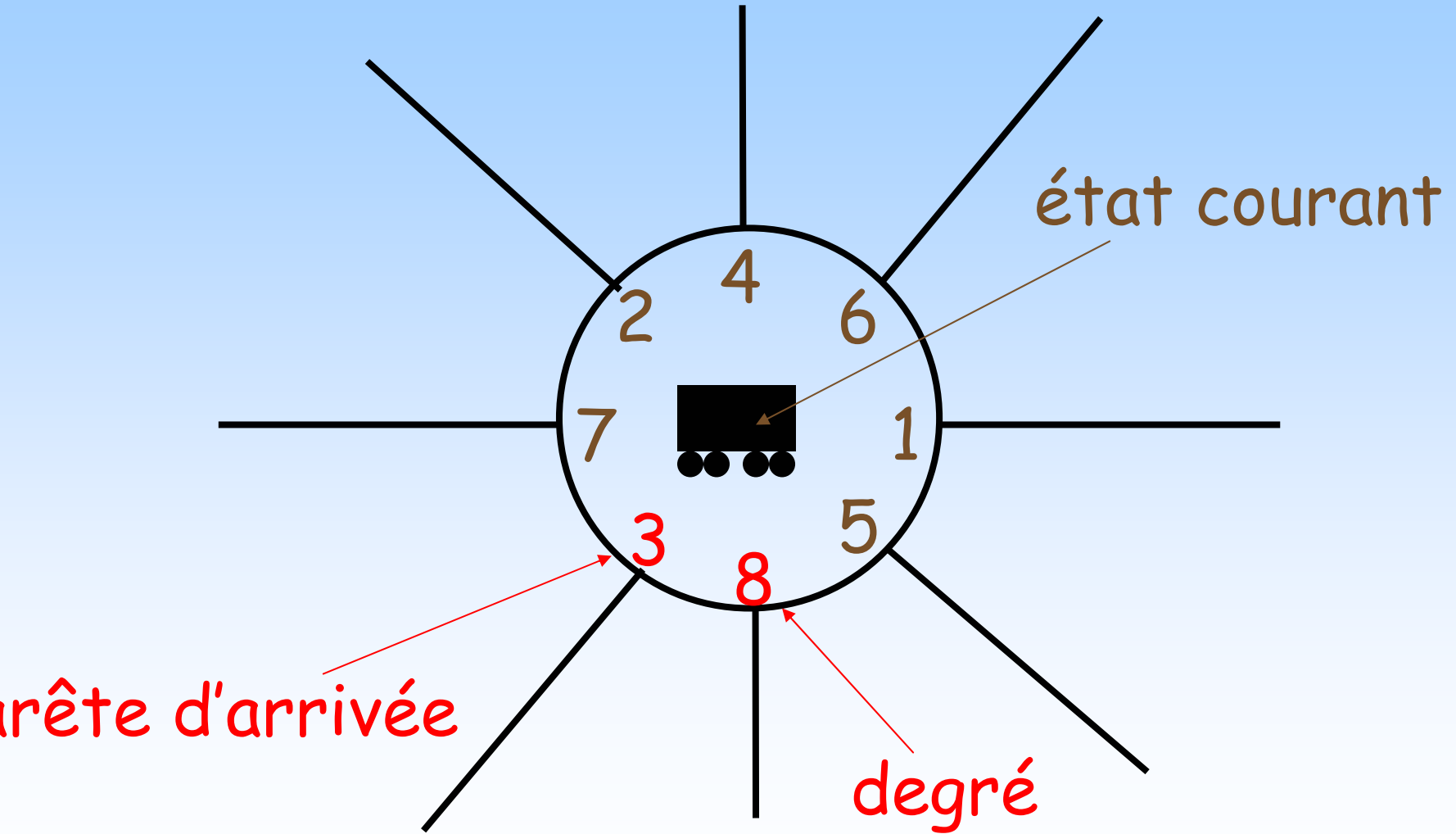
L'agent mobile

- **Automate** déterministe à **K états**
→ robot avec $\leq \log(K)$ bits de mémoire
- Automate de Moore $\{I, P, S, \delta, \pi, s_0\}$
 - I : ensemble des entrées (degré, port)
 - P : ensemble des sorties (port)
 - S : ensemble des états
 - δ : fonction de transition ($I \times A \rightarrow A$)
 - π : fonction de déplacement ($A \rightarrow P$)
 - s_0 : état initial

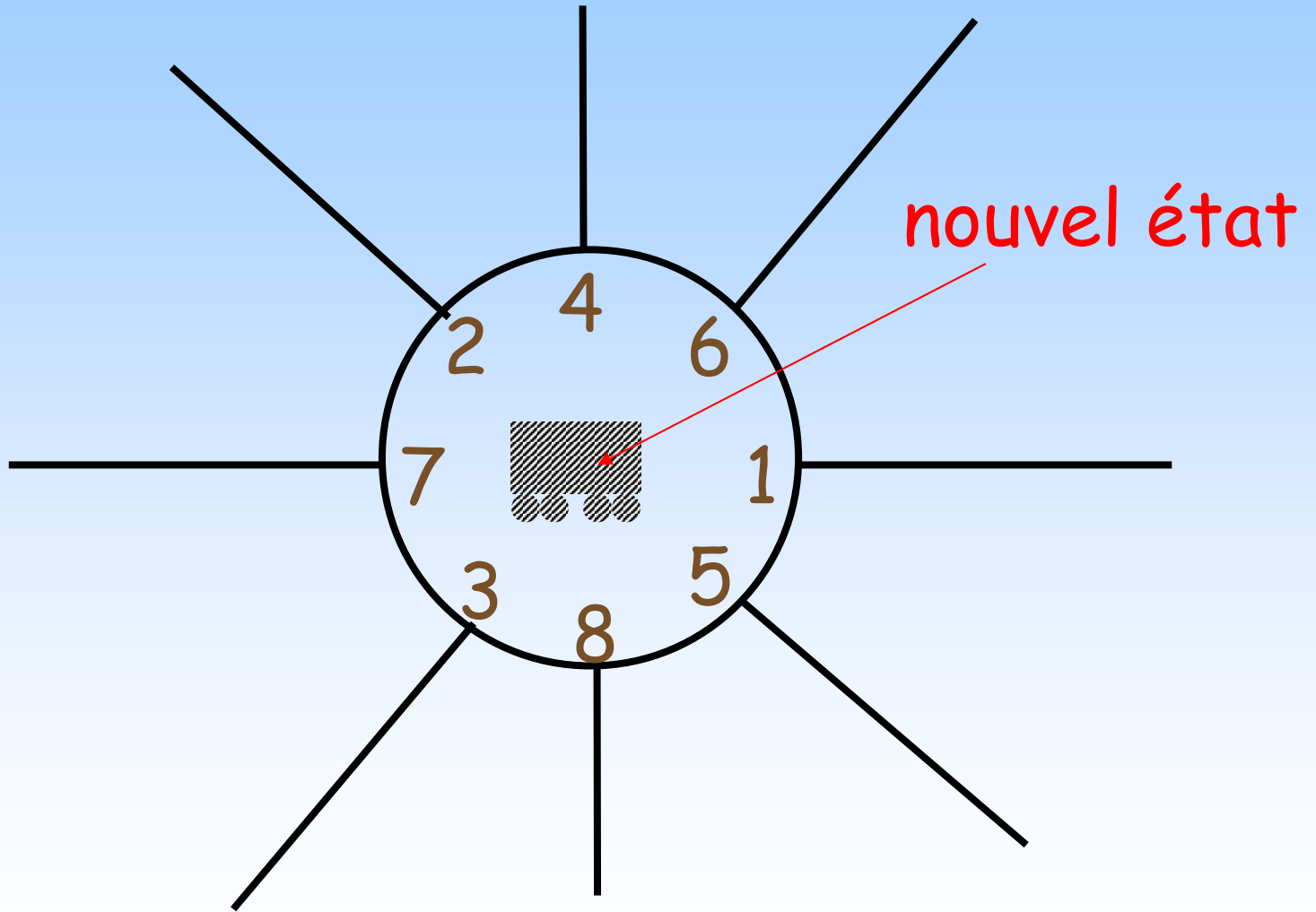
Actions du robot



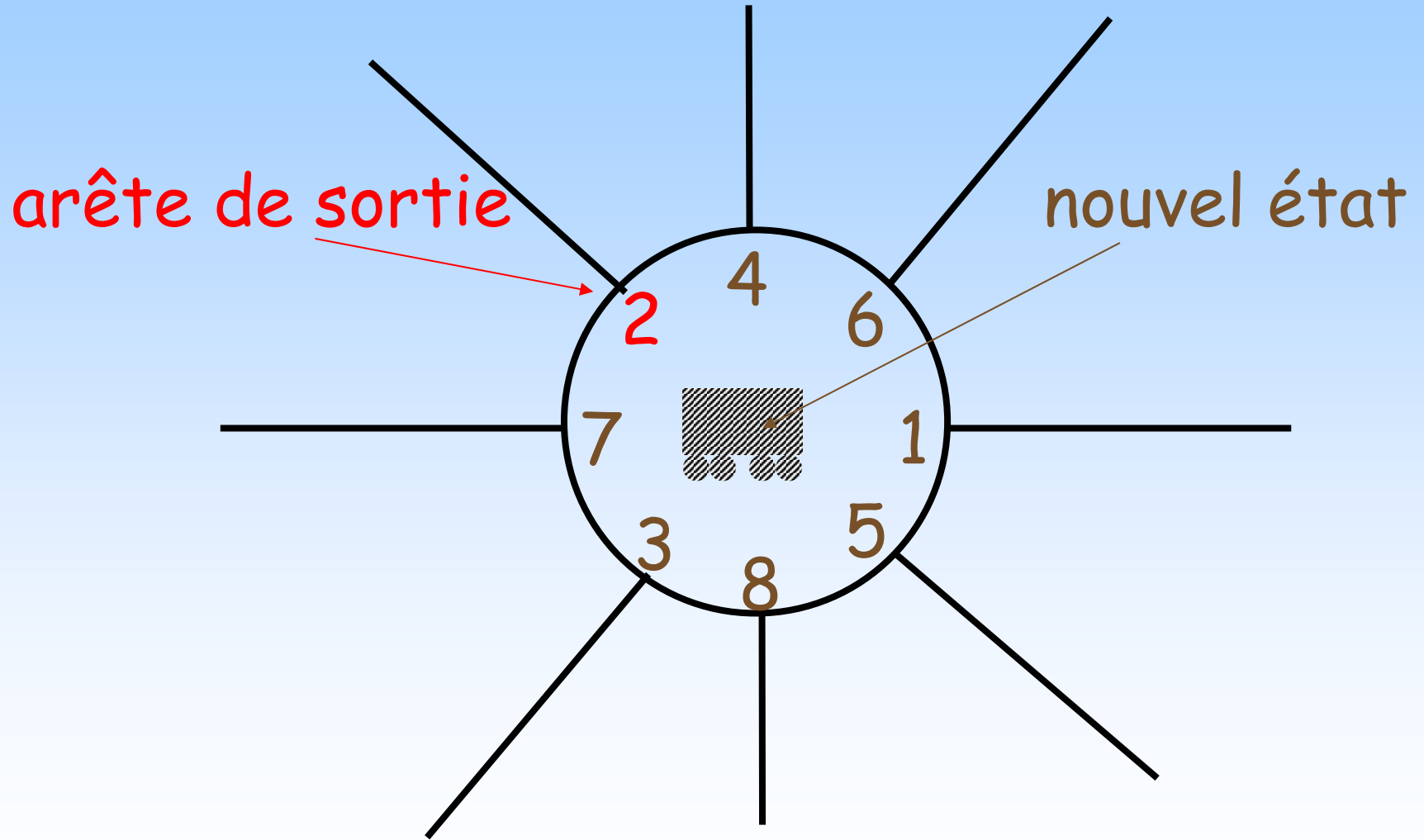
Actions du robot



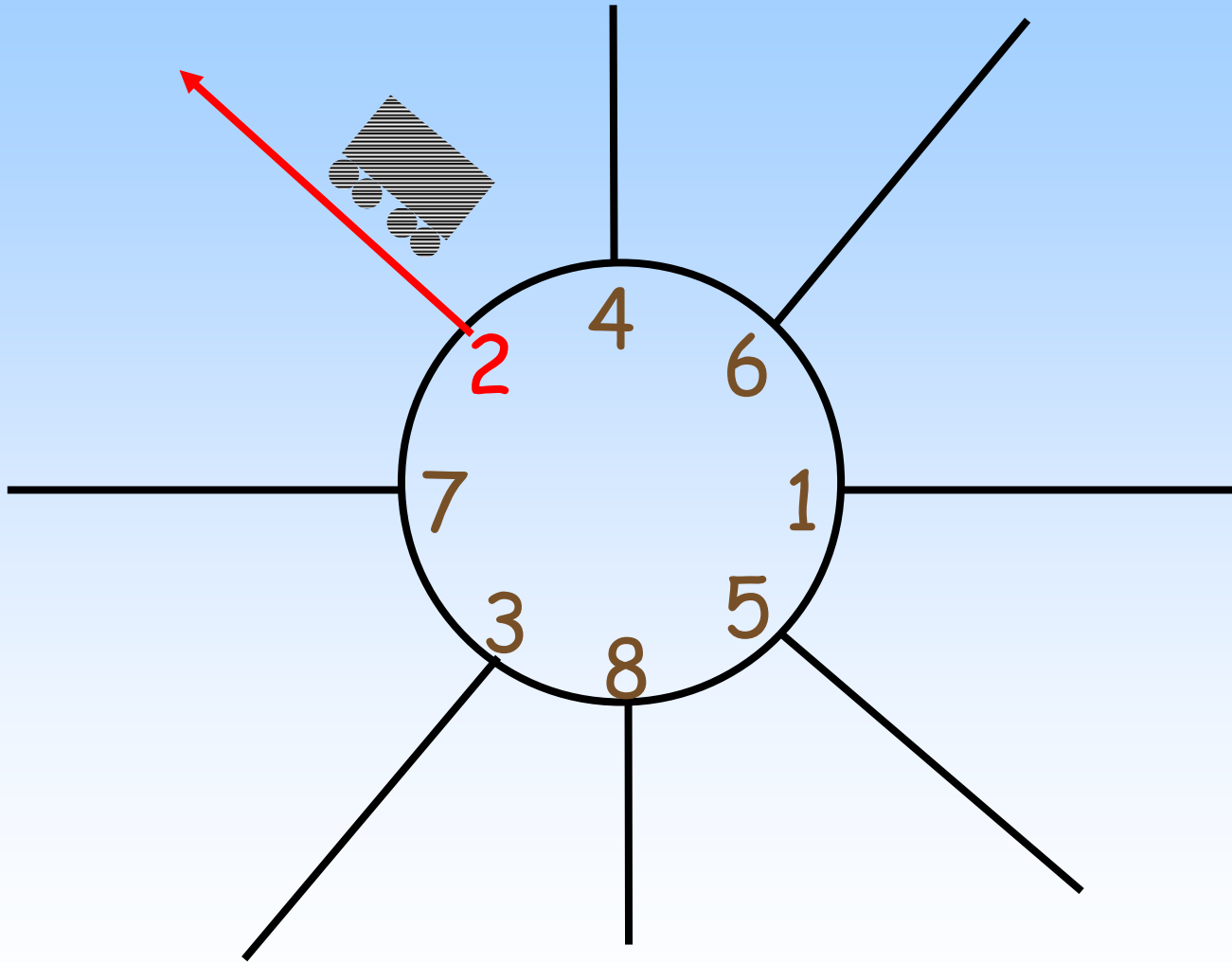
Actions du robot



Actions du robot



Actions du robot



Plan

- Historique
- Construction d'un piège (borne inférieure)
 - piège pour l'automate A = graphe que l'automate A n'explore pas entièrement
- Algorithme optimal (borne supérieure)

Historique (1)

- Première publication sur les labyrinthes
Wiener C. , 1873
- The mouse of Shannon
Shannon, Cl. E., Presentation of a maze-solving machine. In: Cybernetics; Trans. of the 8th Conf. of the Josiah Macy Jr. Foand. (Editor: H. v. Foerster); 173-180 (**1951**)
- Communication orale
Rabin 1966
- Problème du piège
Döpp, K., Automaten in Labyrinthen, EIK 7/2, 79-94;3, 167-189 (**1971**)

Historique (2)

- Premier piège (graphes planaires)
Budach, L., Automata and Labyrinths,
Math. Nachrichten 86, 195-282, 1978
- Résultat le plus récent : piège à $K+1$
sommets pour $d > 3$ et à $2K+1$ sommets
pour $d=3$
Fraigniaud P., Peer G. Pelc A., Peleg D.,
non publié, 2003

Construction d'un piège

- Définition
 - Un graphe G est un **piège** pour un automate A si A n'arrive pas à explorer G .
- Théorème
 - Pour tout automate fini à K états, il existe un piège de **degré $d > 2$** ayant au plus **$K+1$** sommets.

Définitions

Définitions

- Pseudo-palindrome
 - \emptyset , pLp, ppL, Lpp

Définitions

- Pseudo-palindrome
 - \emptyset , pLp, ppL, Lpp
 - Exemple : 122331

Définitions

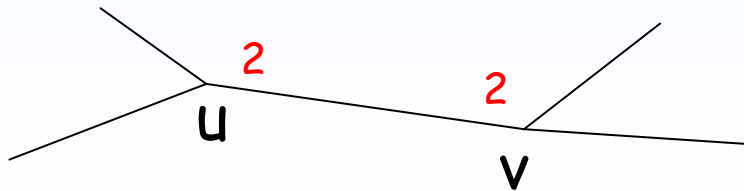
- Pseudo-palindrome
 - \emptyset , pLp, ppL, Lpp
 - Exemple : 122331
 - Palindrome de taille paire : 2112

Définitions

- Pseudo-palindrome
 - \emptyset , pLp, ppL, Lpp
 - Exemple : 122331
 - Palindrome de taille paire : 2112
 - pp-réduction : 3121123122331 \rightarrow 313

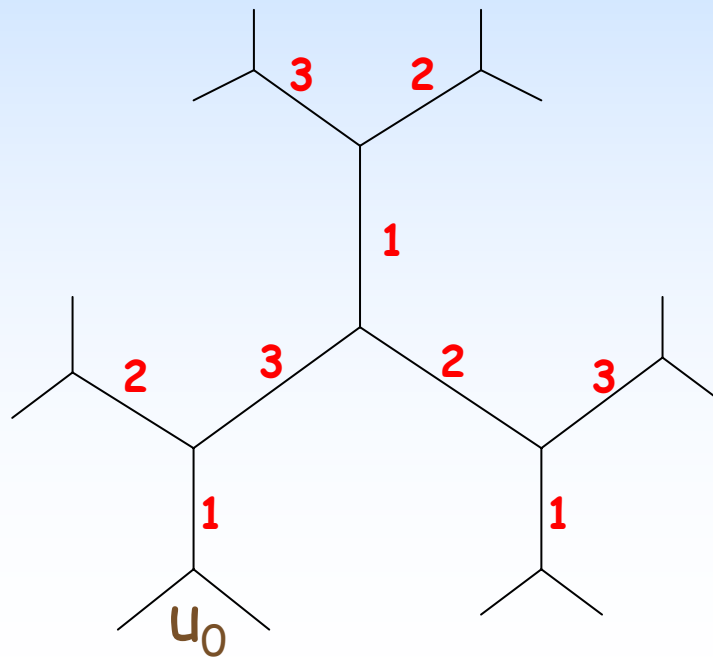
Définitions

- Pseudo-palindrome
 - \emptyset , pLp, ppL, Lpp
 - Exemple : **122331**
 - Palindrome de taille paire : **2112**
 - pp-réduction : **3121123122331** \rightarrow 313
- Graphe à arêtes colorées
 - chaque extrémité de l'arête est accessible par le même numéro de port



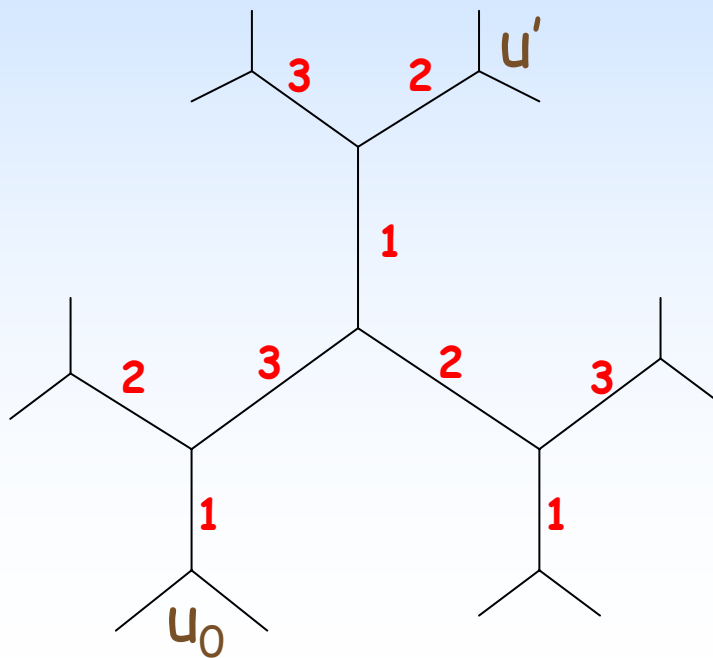
Squelette du piège (1)

- Objectif : piéger le robot dans un cycle
- Graphe de départ : arbre infini de degré 3, à arêtes colorées



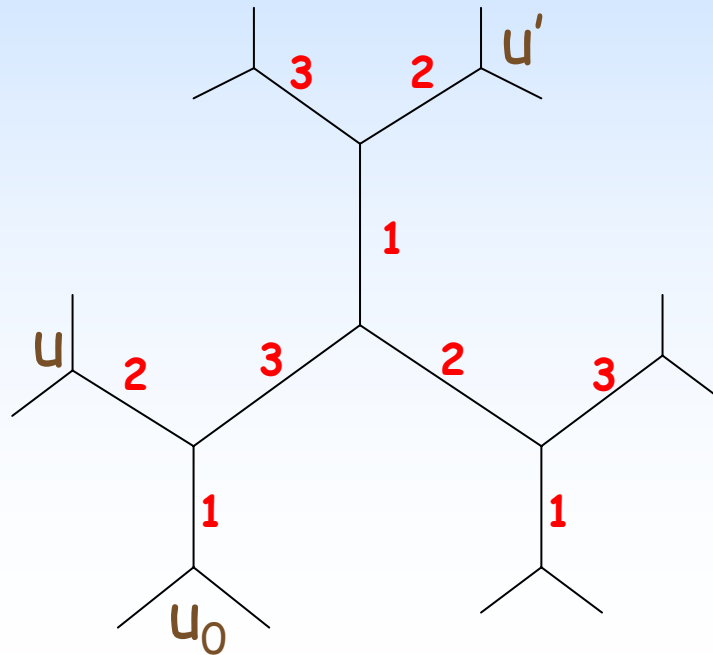
Squelette du piège (1)

- Objectif : piéger le robot dans un cycle
- Graphe de départ : arbre infini de degré 3, à arêtes colorées



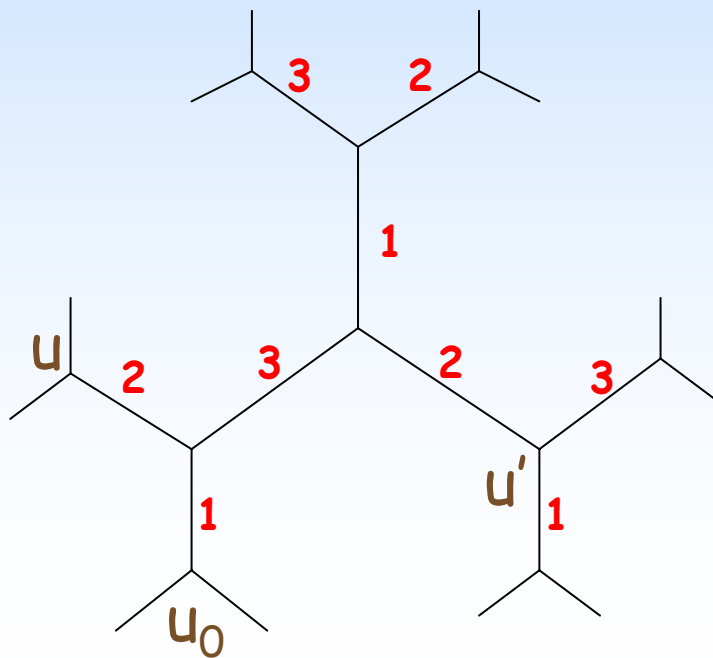
Squelette du piège (1)

- Objectif : piéger le robot dans un cycle
- Graphe de départ : arbre infini de degré 3, à arêtes colorées



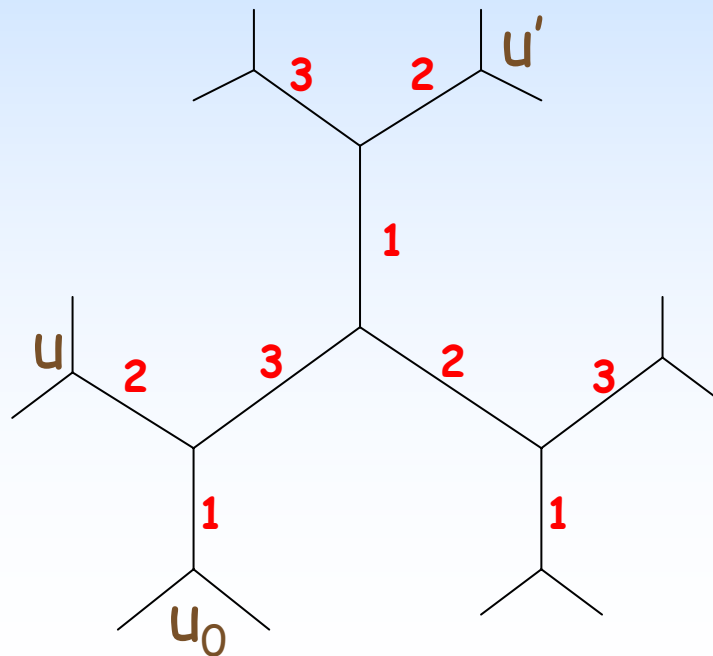
Squelette du piège (1)

- Objectif : piéger le robot dans un cycle
- Graphe de départ : arbre infini de degré 3, à arêtes colorées



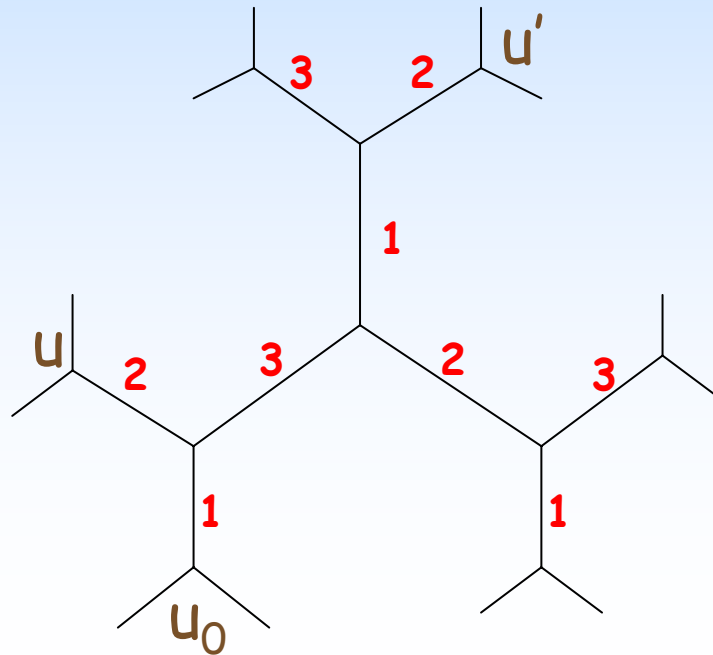
Squelette du piège (1)

- Objectif : piéger le robot dans un cycle
- Graphe de départ : arbre infini de degré 3, à arêtes colorées



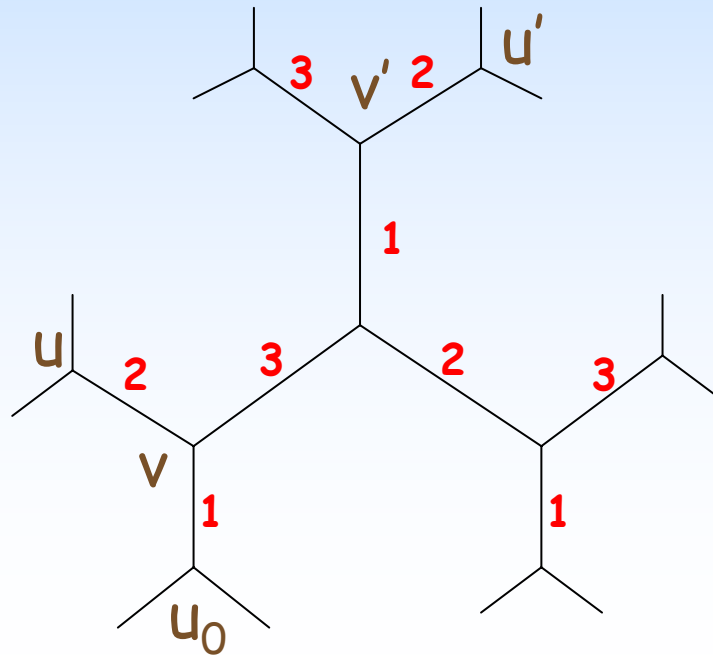
Squelette du piège (2)

- Cas où la pp-réduction n'est pas un palindrome
- Exemple : **1332-23233212** \rightarrow **2312** \rightarrow 31



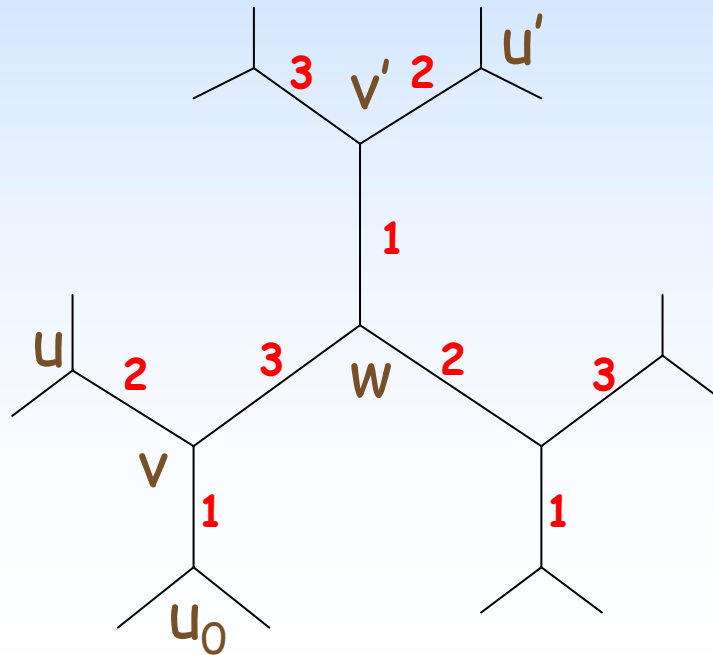
Squelette du piège (2)

- Cas où la pp-réduction n'est pas un palindrome
- Exemple : **1332-23233212** \rightarrow **2312** \rightarrow 31



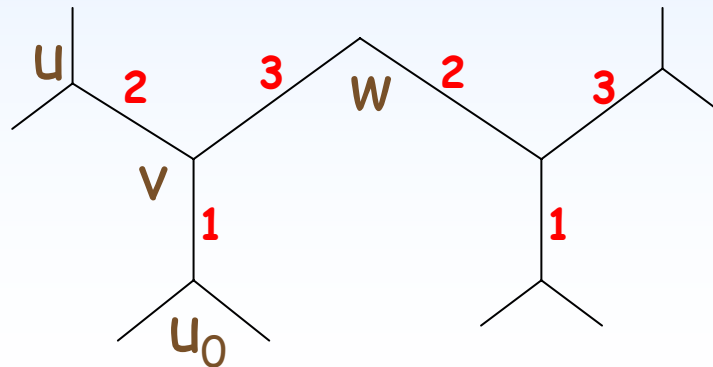
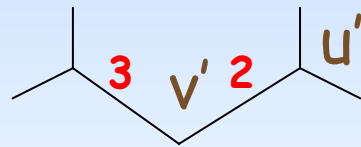
Squelette du piège (2)

- Cas où la pp-réduction n'est pas un palindrome
- Exemple : **1332-23233212** \rightarrow **2312** \rightarrow 31



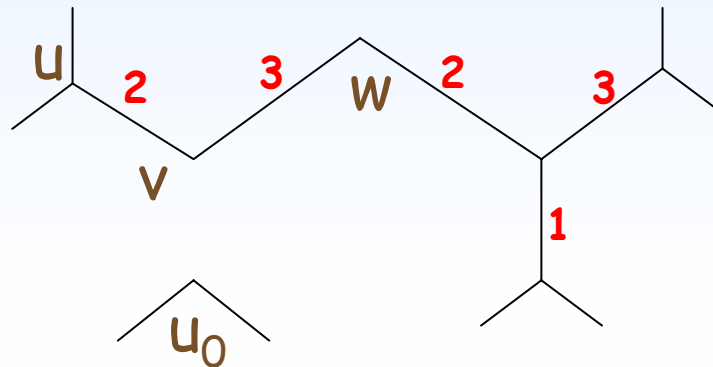
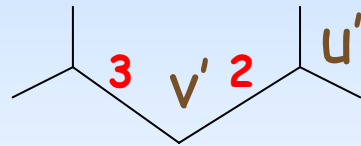
Squelette du piège (2)

- Cas où la pp-réduction n'est pas un palindrome
- Exemple : **1332-23233212** \rightarrow **2312** \rightarrow 31



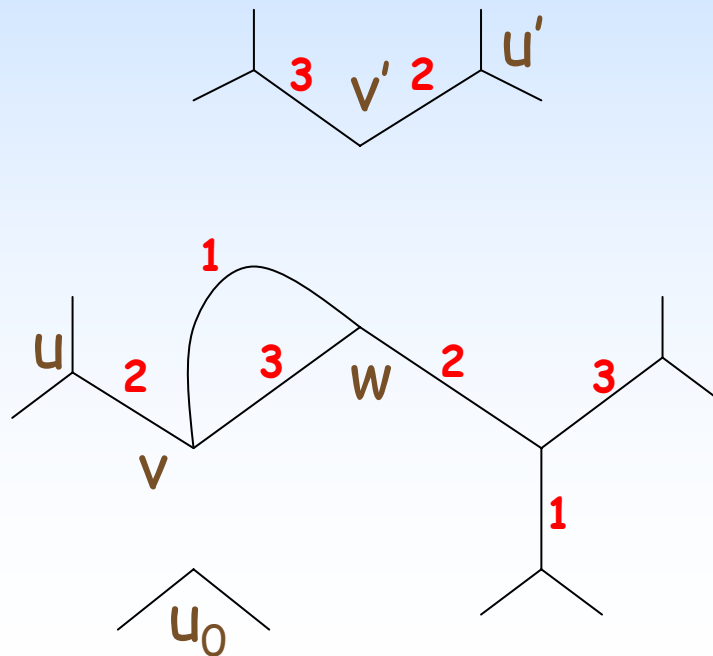
Squelette du piège (2)

- Cas où la pp-réduction n'est pas un palindrome
- Exemple : 1332-23233212 → 2312 → 31



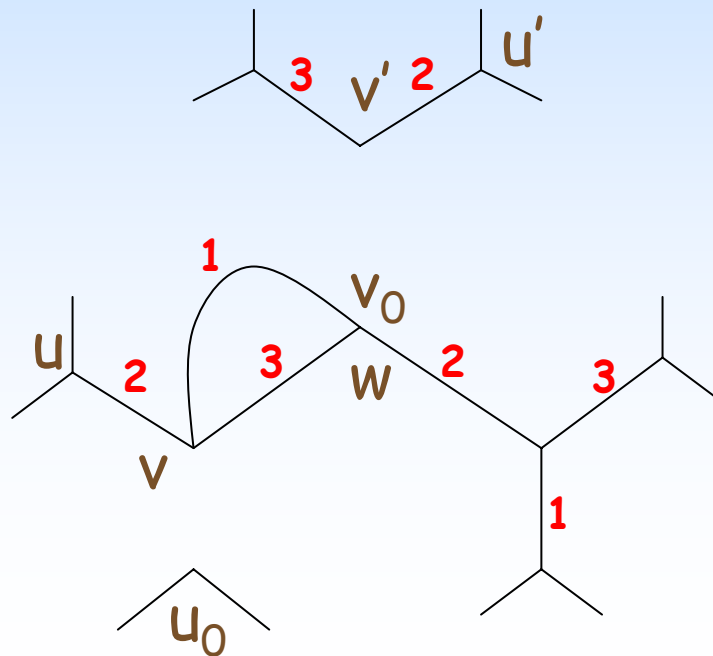
Squelette du piège (2)

- Cas où la pp-réduction n'est pas un palindrome
- Exemple : **1332-23233212** \rightarrow **2312** \rightarrow 31



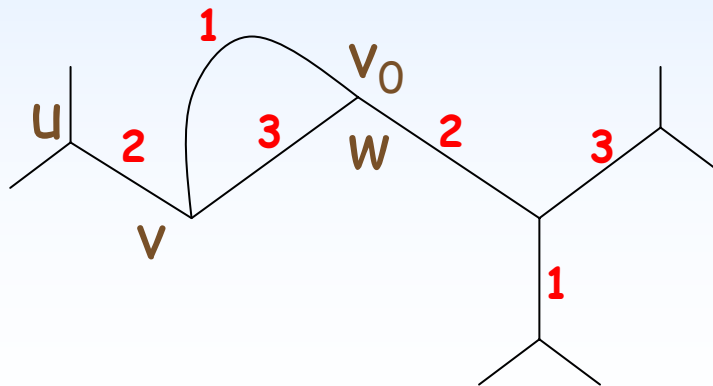
Squelette du piège (2)

- Cas où la pp-réduction n'est pas un palindrome
- Exemple : 1332-23233212 → 2312 → 31



Squelette du piège (2)

- Cas où la pp-réduction n'est pas un palindrome
- Exemple : **1332-23233212** \rightarrow **2312** \rightarrow 31

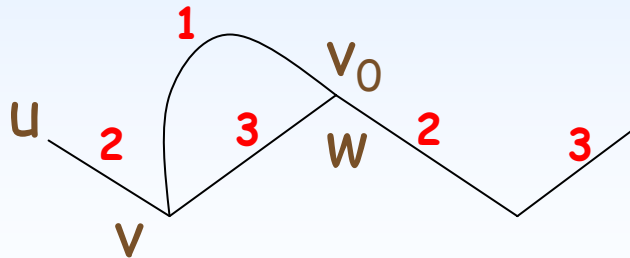


Résultat

- Il existe un piège de $K+1$ sommets ou moins

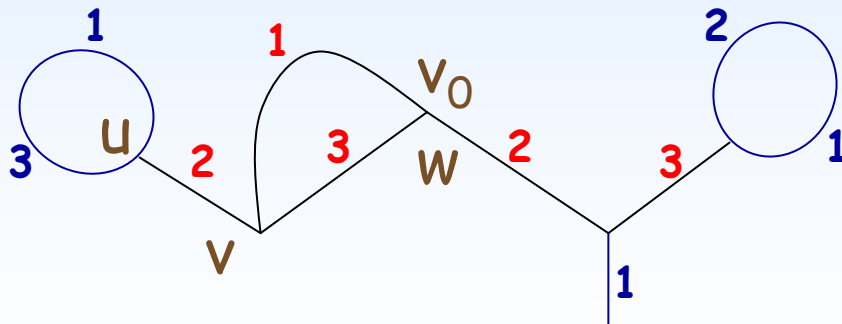
Résultat

- Il existe un piège de $K+1$ sommets ou moins
- Exemple : **1332-23233212**



Résultat

- Il existe un piège de $K+1$ sommets ou moins
- Exemple : **1332-23233212**



Algorithme d'exploration

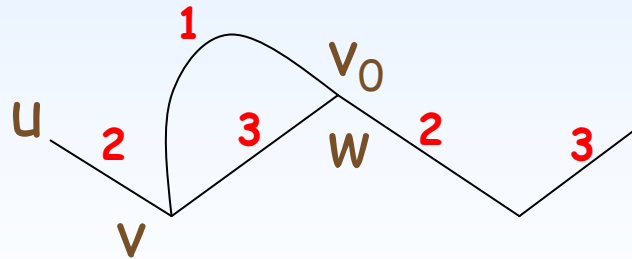
- Théorème
 - Un robot avec $O(D \log d)$ bits de mémoire peut explorer tous les graphes de degré maximum d et de diamètre au plus D
- Preuve
 - Parcours en profondeur d'abord (DFS) de profondeur croissante
 - Mémoire : pile des numéros de port menant à la racine $\rightarrow D \log(d)$ bits

Piège plus approprié

- Il existe un piège de diamètre au plus $4\lceil \log_{d-1}(K) \rceil + 2$

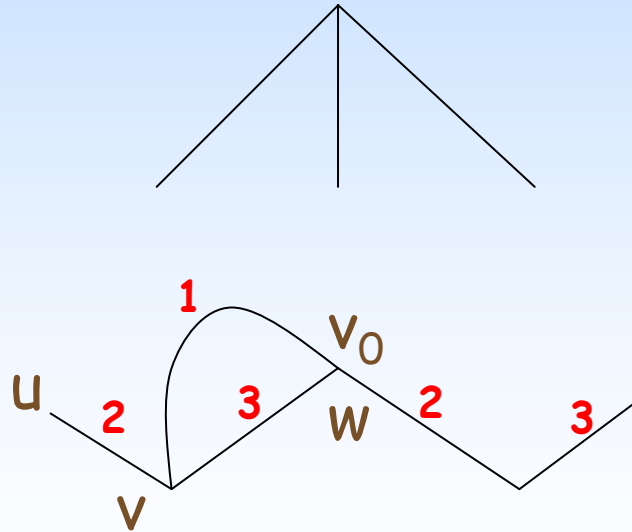
Piège plus approprié

- Il existe un piège de diamètre au plus $4\lceil \log_{d-1}(K) \rceil + 2$
- Exemple : **1332-23233212**



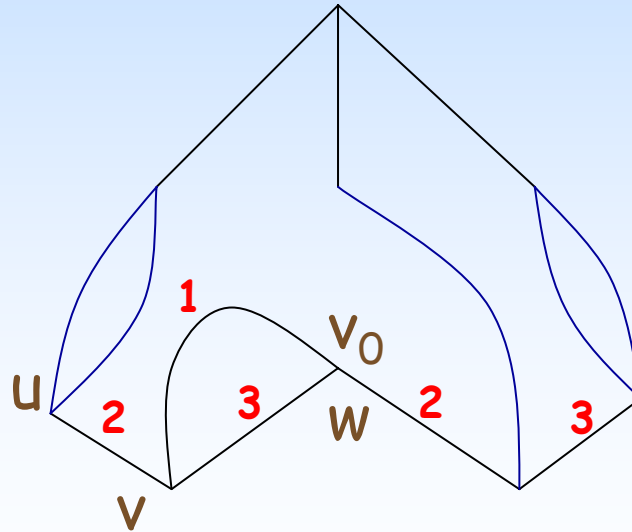
Piège plus approprié

- Il existe un piège de diamètre au plus $4\lceil \log_{d-1}(K) \rceil + 2$
- Exemple : **1332-23233212**



Piège plus approprié

- Il existe un piège de diamètre au plus $4\lceil \log_{d-1}(K) \rceil + 2$
- Exemple : **1332-23233212**



Borne inférieure

- Théorème

- Pour tout automate fini à K états, il existe un piège de degré $d > 2$ de diamètre au plus $4\lceil \log_{d-1}(K) \rceil + 2$.

- Théorème

- Un robot qui peut explorer tous les graphes de degré maximum d et de diamètre au plus D nécessite $\Omega(D \log d)$ bits de mémoire.

Conclusion

- Résultats
 - Piège de $K+1$ sommets pour $d > 2$
 - Optimalité de la mémoire
 - Résultats valables aussi en planaire
- Perspectives
 - Exploration perpétuelle par plusieurs automates
 - Exploration avec arrêt : nécessité d'un marqueur
 - Trouver un piège pour un automate avec un marqueur