

La recherche heuristique

Faculté de sciences et de génie
Département d'informatique et de génie logiciel
IFT-2003 Intelligence artificielle I
Laurence Capus



Plan

- Qu'est-ce qu'une heuristique ?
- Qu'est-ce qu'une technique de recherche heuristique ?
- Exemples de techniques de recherche heuristique
- Exemples de techniques de recherche heuristique spécifiques aux jeux

Introduction

- Pour rechercher une solution dans un espace d'états, on peut balayer tout l'espace avec une technique de recherche exhaustive. Ceci n'est pas très efficace.
- Des techniques ont été mises au point pour améliorer l'efficacité en se basant sur des informations propres au contexte du problème appelées connaissances.
- Par exemple, un joueur qui a de l'expérience dans un jeu va être capable d'estimer si un état est plus ou moins proche de la solution et donc de choisir le meilleur coup à jouer. Le joueur utilise dans ce cas ses connaissances du jeu acquises en jouant.

Notion d'heuristique

- Heuristique
 - Discipline qui se propose de dégager les règles de la recherche scientifique et de la découverte (Petit Larousse).
 - Dans le contexte de la recherche dans un espace d'états, il s'agit d'une **technique pour choisir l'opération la plus susceptible de mener à une solution acceptable.**

Notion d'heuristique

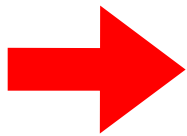
- Fonction heuristique
 - Estimé du nombre d'états à traverser pour passer de l'état courant à l'état désiré
 - Estimé de la *valeur* d'un état

Notion d'heuristique

- Règle heuristique
 - « Règle du pouce »
 - Règle permettant de choisir une bonne opération à appliquer sans effectuer de recherche exhaustive dans l'espace d'états
 - Résolution de conflits
 - Si plusieurs règles sont applicables dans un certain état, on peut exploiter le fait que les règles sont examinées séquentiellement en mettant les « meilleurs » choix en premier.

Notion d'heuristique

- Choisir une opération
 - Avec une fonction heuristique :
 1. Calculer la valeur heuristique de chaque état qu'il est possible d'atteindre à partir de l'état courant
 2. Choisir l'opération menant à l'état ayant la meilleure valeur heuristique (la plus élevée ou la moins élevée selon la fonction heuristique)
 - Avec des règles heuristiques :
 1. Choisir l'opération suggérée par la première règle heuristique dont les conditions sont rencontrées dans l'état courant.

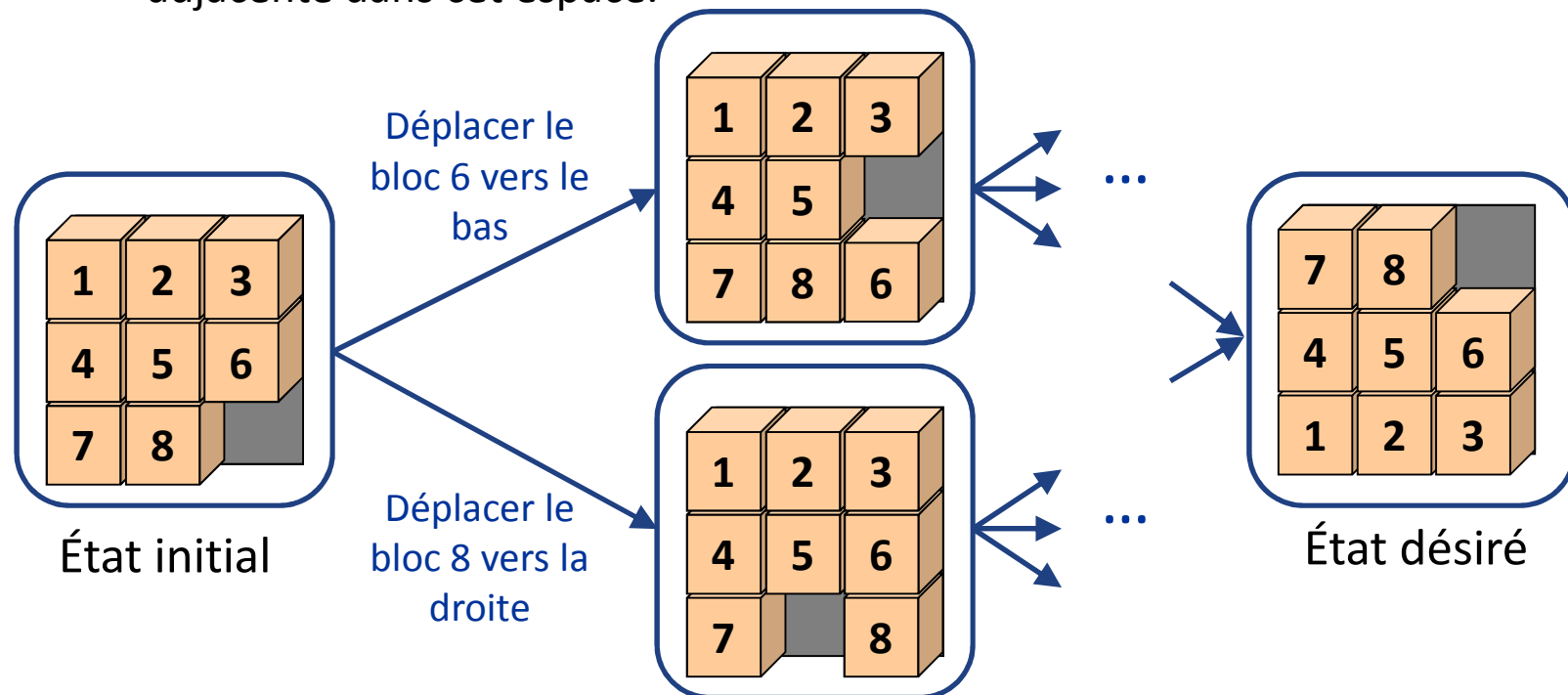


L'ordonnancement des règles fait partie de l'heuristique

Notion d'heuristique

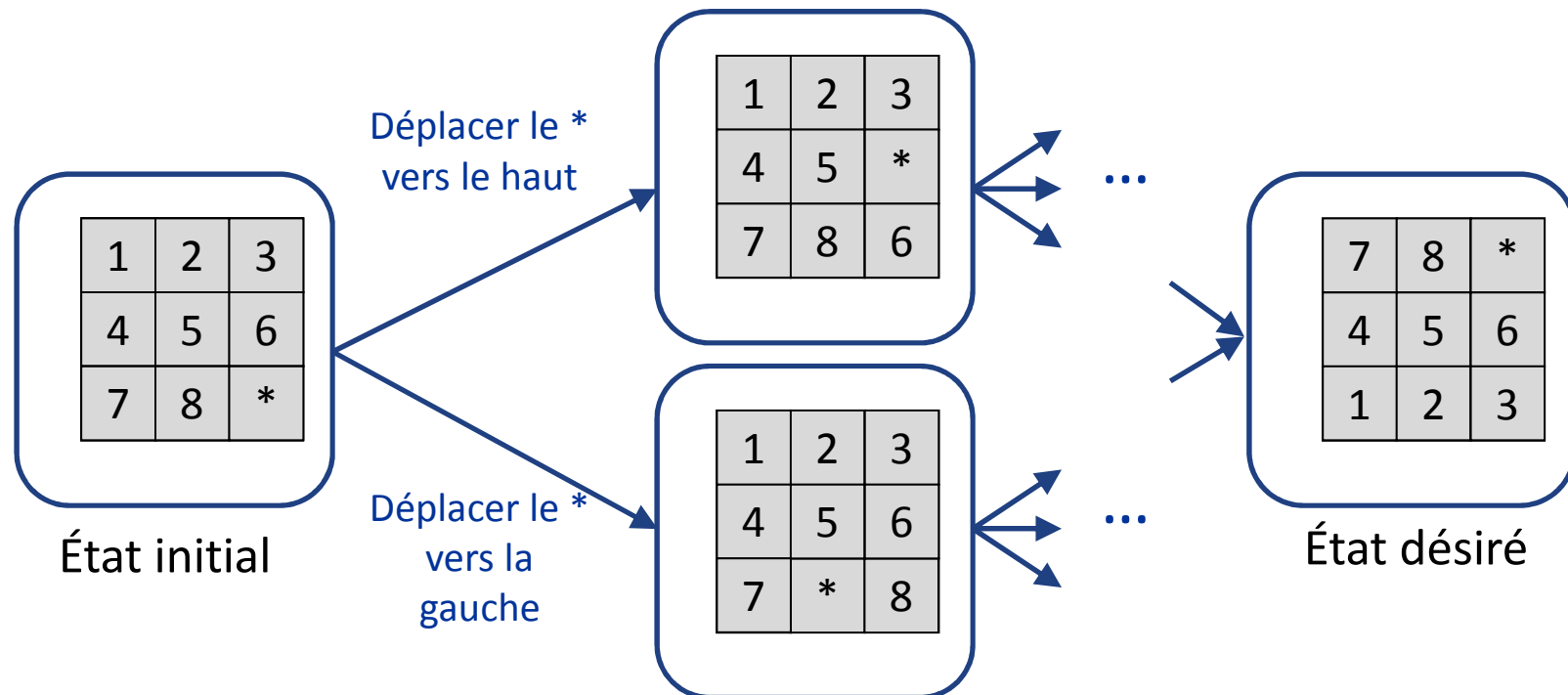
- Exemple : le jeu du taquin

- Le jeu consiste à déplacer des tuiles dans un cadre de 3x3 positions jusqu'à obtenir une configuration donnée
- Il y a 8 tuiles et un espace vide. L'espace vide permet de déplacer une tuile adjacente dans cet espace.



Notion d'heuristique

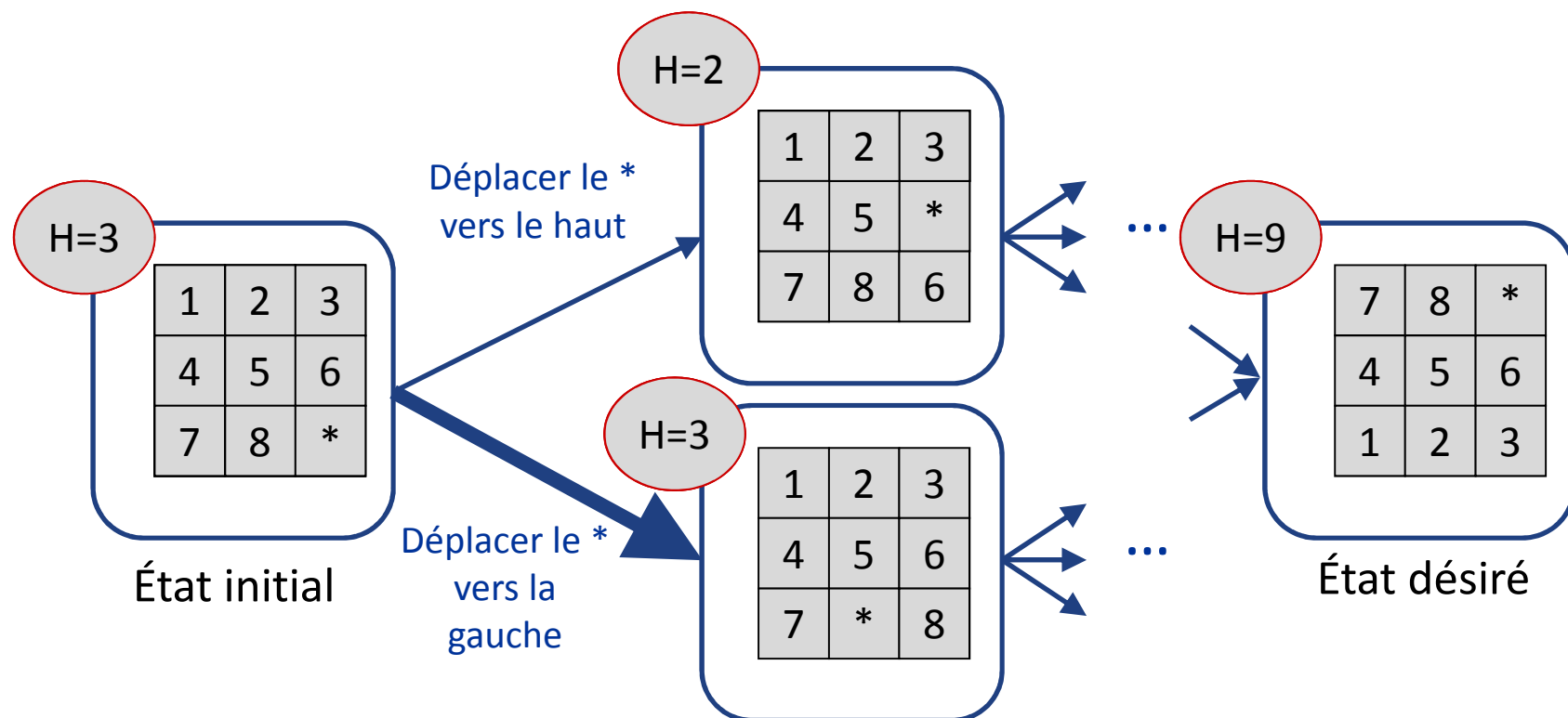
- Exemple : le jeu du taquin
 - On peut représenter un état par un tableau de 3x3 cases.
 - On peut représenter l'espace vide comme une case particulière (notée *)
 - Les opérations peuvent être le déplacement de la case vide.



Notion d'heuristique

- Exemple : le jeu du taquin

- Exemple de fonction heuristique pour l'évaluation d'un état : **H = le nombre de tuiles qui sont à la bonne place**. On préférera donc les états où H est plus élevé.



Exemples de techniques

- Technique du meilleur d'abord
- Technique A*

Technique du meilleur d'abord

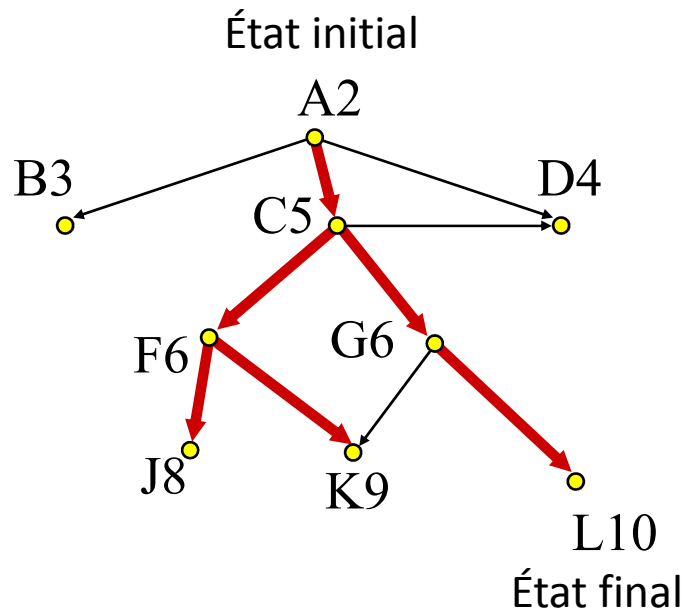
- C'est une combinaison des techniques en largeur et en profondeur, basée sur les avantages des deux techniques.
- La recherche en **profondeur** est intéressante car elle permet de trouver une solution sans avoir à faire l'expansion de toutes les branches candidates. La recherche en **largeur**, elle, est intéressante car elle ne risque pas de nous conduire dans des culs-de-sac. Une façon de combiner les deux consiste à suivre un seul chemin à la fois, mais d'en changer lorsqu'il s'en présente un de plus prometteur.
- À chaque étape du processus de recherche du **meilleur d'abord**, on sélectionne le nœud le plus prometteur de tous les nœuds générés jusque-là. On réalise ceci en appliquant une fonction heuristique appropriée à chacun d'entre eux. On réalise ensuite l'expansion du nœud en utilisant les règles applicables. Parmi les successeurs obtenus, si l'un d'entre eux est une solution on peut quitter. Sinon, tous les nouveaux nœuds sont ajoutés à l'ensemble des nœuds déjà générés et, de nouveau, le nœud le plus prometteur est choisi pour devenir le nœud courant.

Technique du meilleur d'abord

- Dans l'implantation de cette technique de recherche dans un graphe, nous avons besoin de deux listes de nœuds:
 - OUVERTS : des nœuds qui ont été générés et pour lesquels on a calculé la fonction heuristique mais qui n'ont pas déjà été examinés (n'ont pas eu leurs successeurs générés). OUVERTS est, en fait, une file avec priorité dans laquelle les éléments avec la plus haute priorité sont ceux avec les valeurs les plus prometteuses de la fonction heuristique.
 - FERMÉS : des nœuds qui ont déjà été examinés. Nous devons garder ces nœuds en mémoire si nous voulons faire une recherche dans un graphe plutôt que dans un arbre, puisque chaque fois qu'un nouveau nœud est généré nous devons vérifier s'il n'a pas été généré auparavant.

Technique du meilleur d'abord

- Exemple 1



#	OUVERTS	FERMES
1	[A2]	[]
2	[C5,D4,B3]	[A]
3	[F6,G6,D4,B3]	[A,C]
4	[K9,J8,G6,D4,B3]	[A,C,F]
5	[J8,G6,D4,B3]	[A,C,F,K]
6	[G6,D4,B3]	[A,C,F,K,J]
7	[L10,D4,B3]	[A,C,F,K,J,G]
8	SUCCÈS	[A,C,F,K,J,G,L]

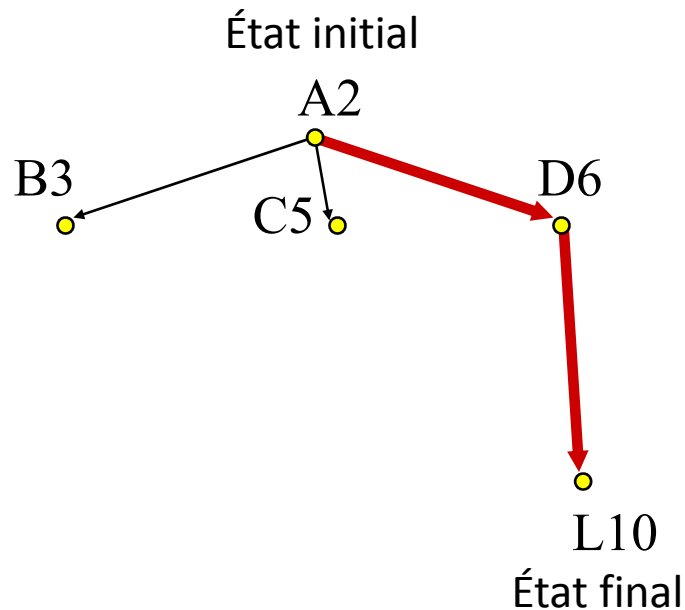
Trajet : A,C,G,L succès

Notes :

- Un état E et sa valeur heuristique N sont notés par EN (p.ex. A2)
- Plus l'état est intéressant, plus la valeur heuristique augmente

Technique du meilleur d'abord

- Exemple 2



#	OUVERTS	FERMES
1	[A2]	[]
2	[D6,C5,B3]	[A]
3	[L10,C5,B3]	[A,D]
4	succès	[A,D,L]

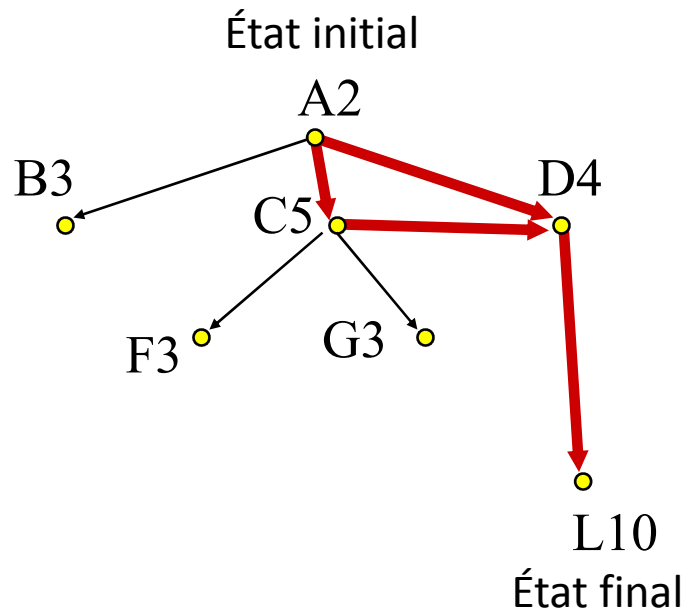
Trajet : A,D,L succès

Notes :

- Un état E et sa valeur heuristique N sont notés par EN (p.ex. A2)
- Plus l'état est intéressant, plus la valeur heuristique augmente

Technique du meilleur d'abord

- Exemple 3



#	OUVERTS	FERMES
1	[A2]	[]
2	[C5,D4,B3]	[A]
3	[D4,B3,F3,G3]	[A,C]
4	[L10,B3,F3,G3]	[A,C,D]
	SUCCÈS	[A,C,D,L]

Trajet : A,D,L succès

Notes :

- Un état E et sa valeur heuristique N sont notés par EN (p.ex. A2)
- Plus l'état est intéressant, plus la valeur heuristique augmente

Technique A*

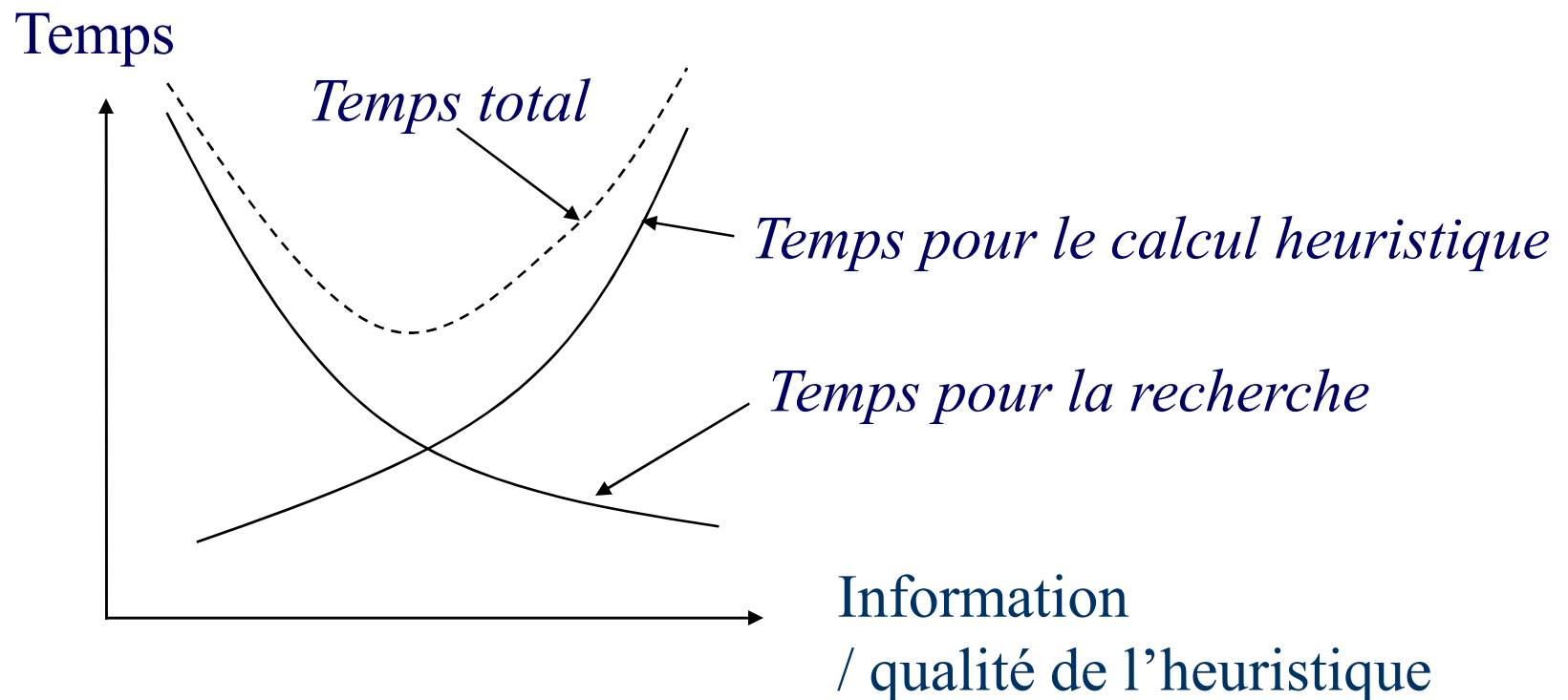
- Méthode améliorée
- Fonction heuristique = somme des deux fonctions **g** et **h**
 - **g** : mesure du coût pour aller de l'état initial à l'état courant (somme exacte des coûts d'application de chacune des règles pour arriver à l'état courant)
 - **h** : estimation du coût additionnel pour aller du nœud courant au nœud solution (liée au domaine du problème)

Technique A*

- Expansion d'un nœud jusqu'à la génération d'un nœud solution
- Procédure de recherche par étapes :
 - Sélection du nœud le plus prometteur parmi les nœuds générés jusque-là mais dont on n'a pas fait l'expansion
 - Génération des successeurs du nœud choisi
 - Application de la fonction heuristique $f=g+h$ à ces successeurs
 - Ajout des nœuds candidats après vérification qu'il n'ont pas été déjà générés

Mesures d'évaluation d'une heuristique

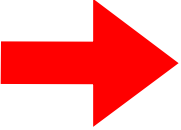
- Compromis entre efficacité et information dans une recherche heuristique



Exemples de techniques spécifiques aux jeux

- Procédure minimax
- Élagage Alpha-Beta

Recherche heuristique pour les jeux

- Beaucoup de jeux consistent en un affrontement entre deux adversaires
 - Exemples : échecs, tic-tac-toe, Othello, Go, Dames, Backgammon, etc.
 - Un joueur ne peut pas contrôler les opérations de son adversaire
 - Un adversaire est hostile et imprévisible
-  **• Un enjeu supplémentaire pour les techniques de recherche**
• Une bonne occasion d'utiliser des heuristiques

Recherche heuristique pour les jeux

- Procédure MINIMAX
 - Les adversaires dans une partie à 2 joueurs sont appelés MIN et MAX.
 - MAX représente le joueur qui tente de gagner
 - MIN représente le joueur qui tente d'empêcher MAX de gagner
 - Hypothèse : MIN possède les mêmes connaissances que MAX sur l'espace d'états et appliquera rationnellement ces connaissances pour gagner la partie.

Recherche heuristique pour les jeux

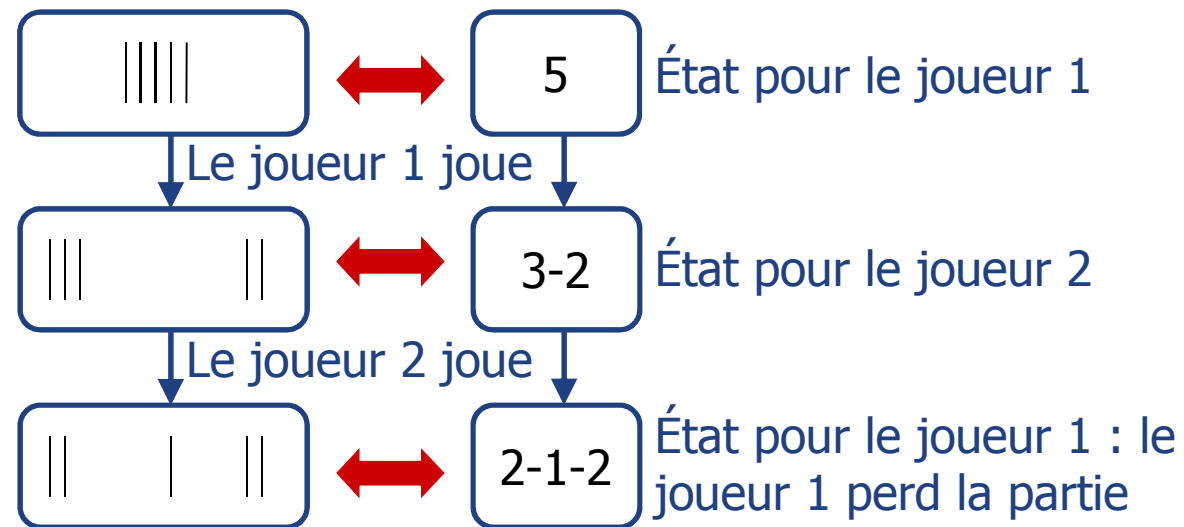
- Procédure MINIMAX

1. Descendre jusqu'aux nœuds les plus profonds (ou jusqu'à une profondeur donnée) et calculer la valeur heuristique de ces nœuds
2. Propager les valeurs vers la racine en respectant les règles suivantes :
 - Si le nœud-parent est un nœud MAX, alors lui donner la valeur maximum parmi celles de ses nœuds-fils
 - Si le nœud-parent est un nœud MIN, alors lui donner la valeur minimum parmi celles de ses nœuds-fils

Recherche heuristique pour les jeux

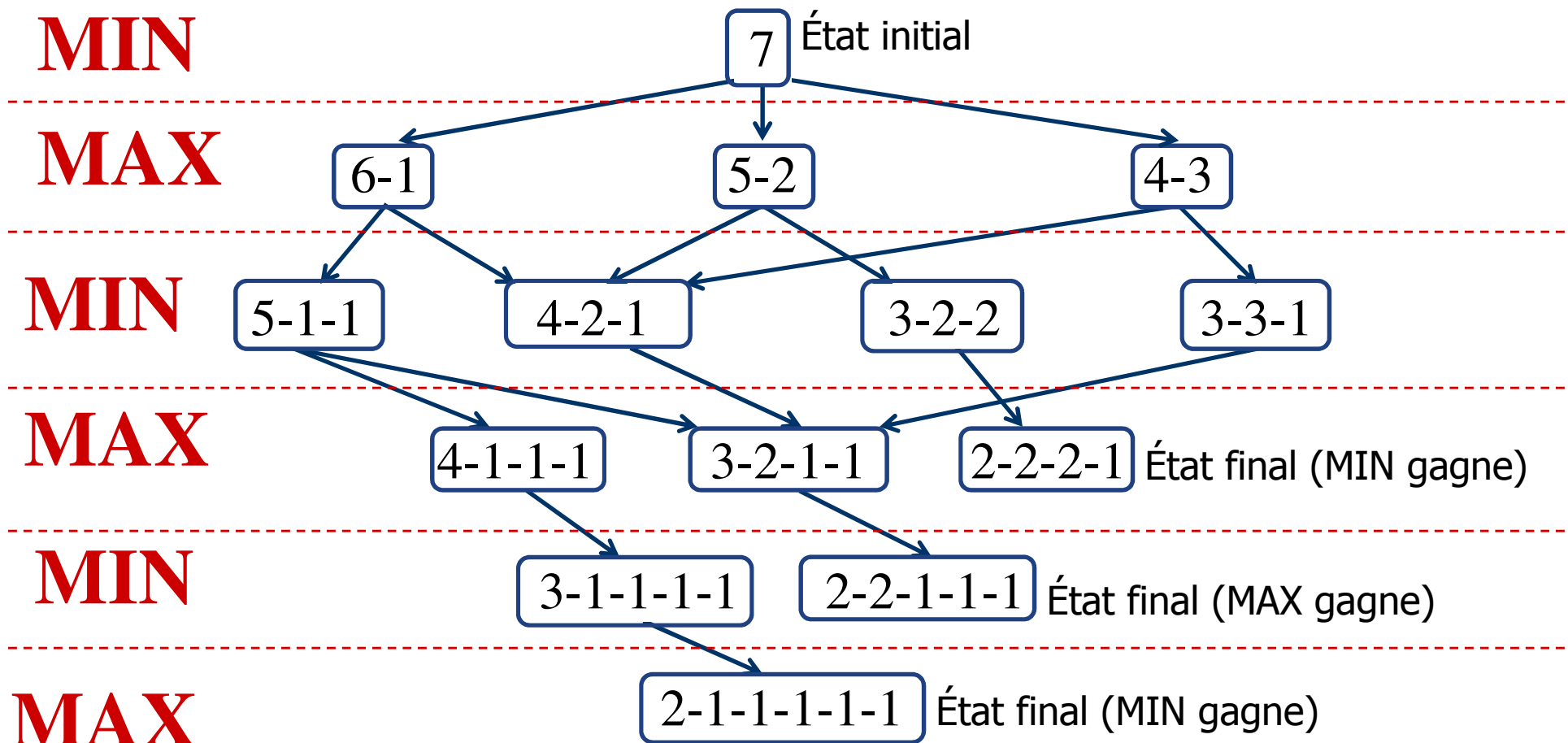
- Exemple 1 : le jeu de NIM

- Initialement, on dispose d'un ensemble de n bâtonnets.
- À chaque tour, un joueur doit séparer un ensemble en 2 sous-ensembles de cardinalités différentes, sinon il perd.
- L'espace d'états est petit si n est petit.
- Ex :



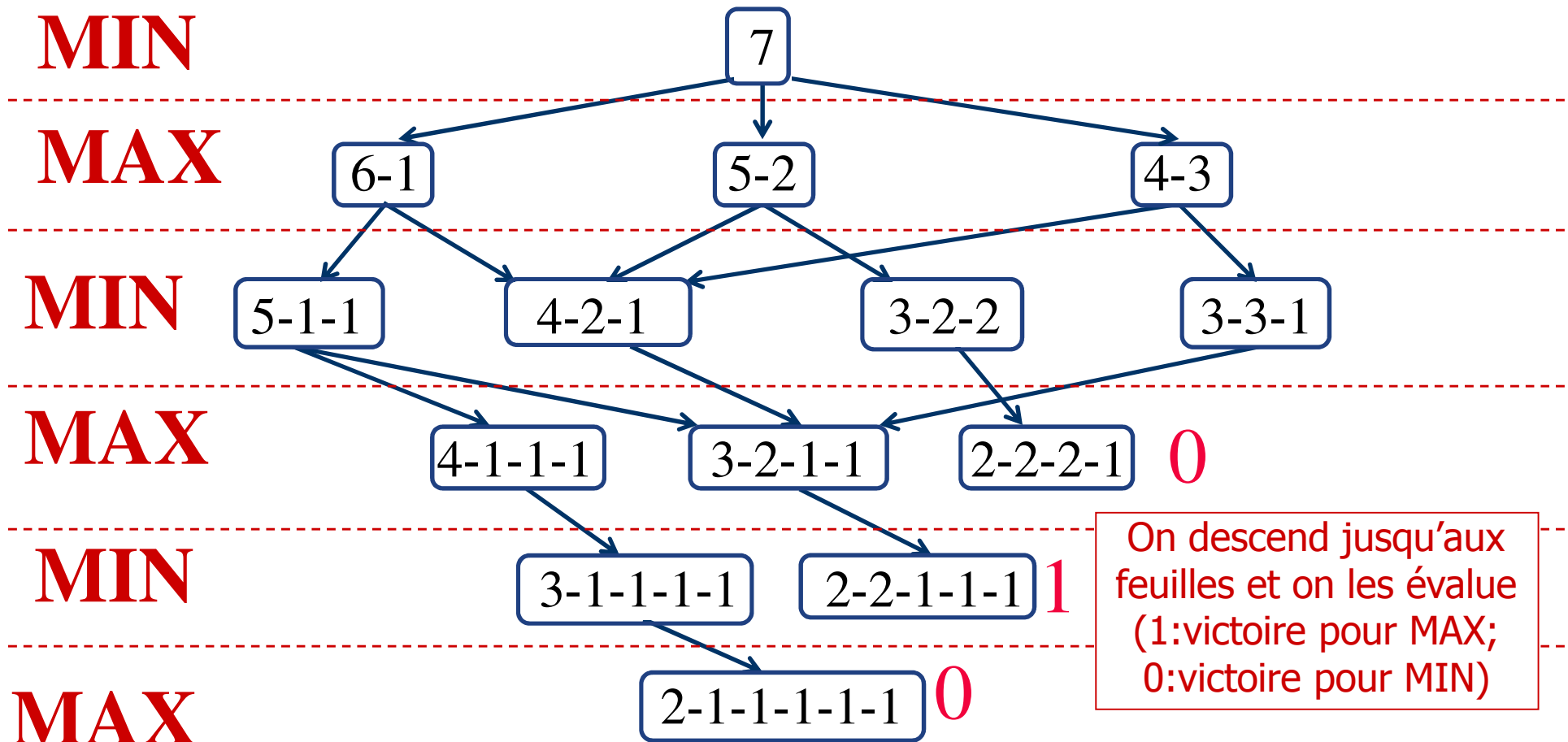
Recherche heuristique pour les jeux

- Exemple 1 : espace d'états pour $n=7$



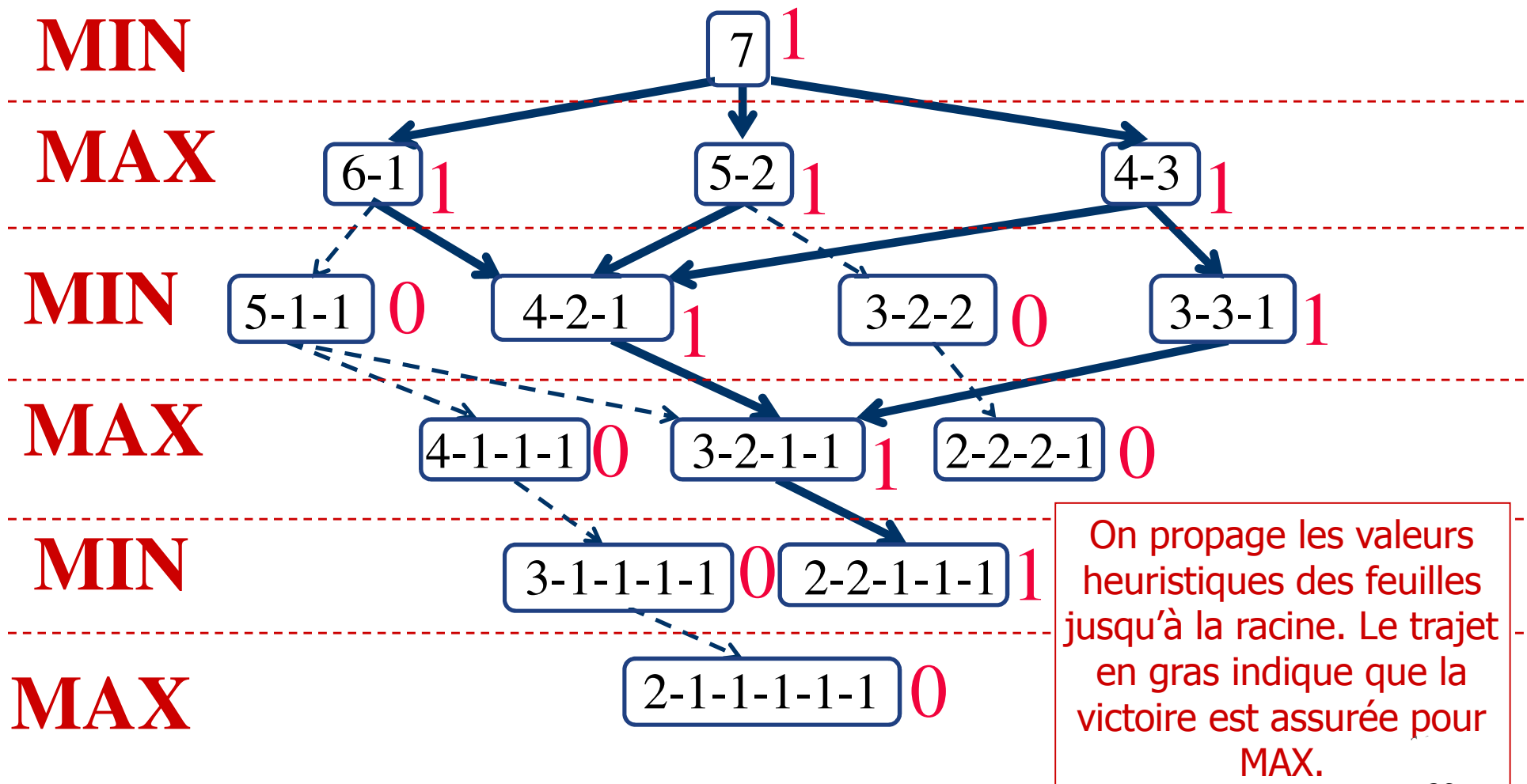
Recherche heuristique pour les jeux

- Exemple 1 : valeurs heuristiques des feuilles



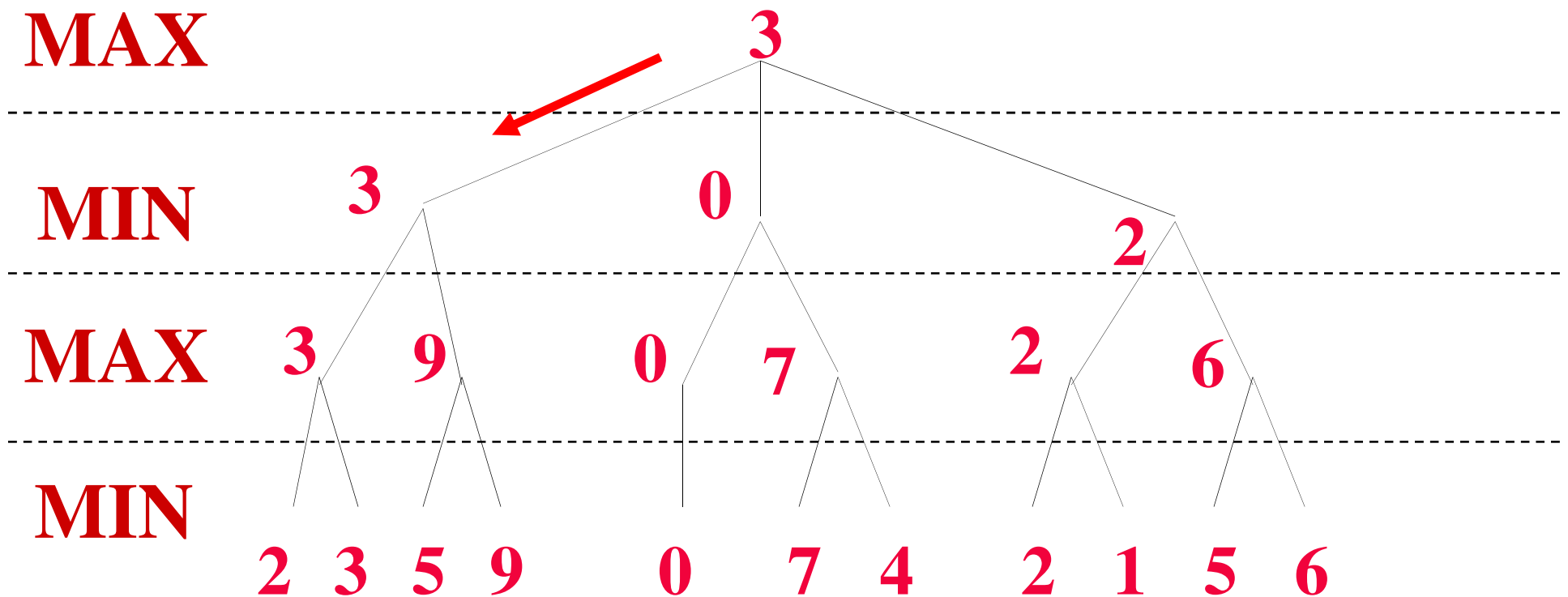
Recherche heuristique pour les jeux

- Exemple 1 : propagation des valeurs heuristiques



Recherche heuristique pour les jeux

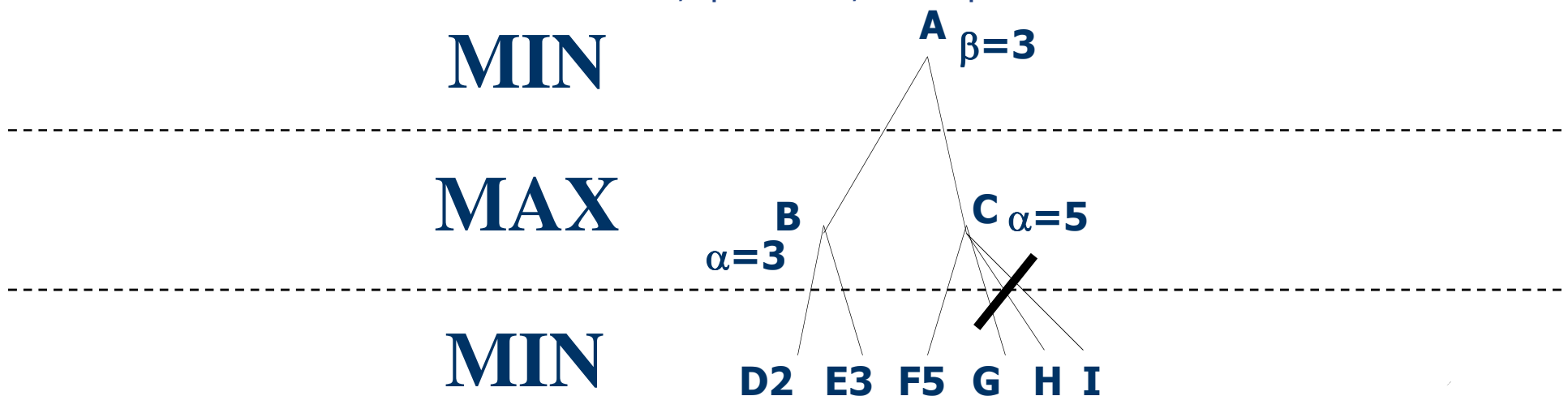
- Exemple 2



Note : Dans cet exemple, un état est meilleur si sa valeur heuristique est plus élevée

Recherche heuristique pour les jeux

- Procédure Alpha-Beta
 - Améliorer l'efficacité de Minimax
 - Construire l'arbre en profondeur d'abord en ne considérant pas certaines branches de l'arbre de recherche.
 - Exemple
 - Plus la valeur heuristique augmente, plus l'état est intéressant pour MAX
 - Inutile de considérer les nœuds G, H et I puisqu'au nœud A, on suppose que MIN choisira le nœud B, qui vaut 3, alors que le nœud C vaut au moins 5.

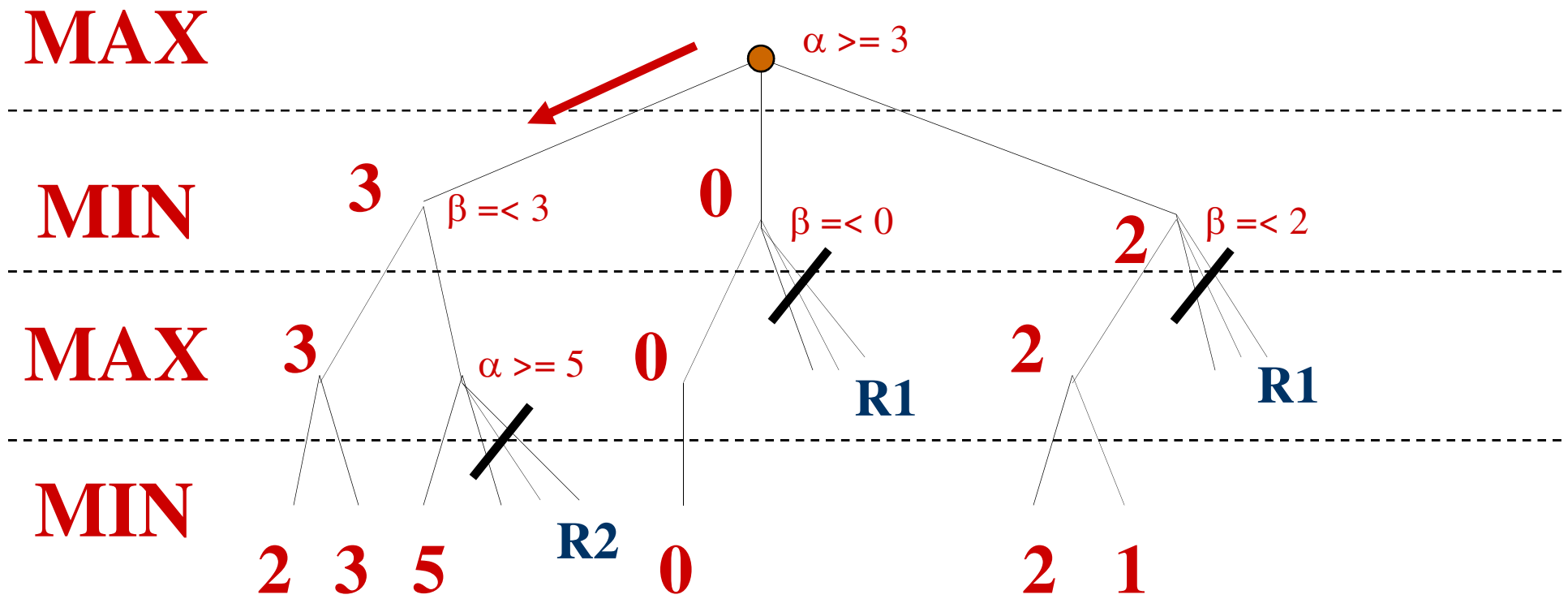


Recherche heuristique pour les jeux

- Procédure Alpha-Beta
 - 2 valeurs, appelées Alpha et Beta, sont créées durant la recherche :
 - Alpha est associée aux nœuds MAX et ne peut jamais diminuer
 - Beta est associée aux nœuds MIN et ne peut jamais augmenter
 - 2 règles pour élaguer l'arbre de recherche :
 - R1: La recherche peut être stoppée sous tout nœud MIN ayant une valeur Beta inférieure ou égale à la valeur Alpha de n'importe lequel de ses ancêtres MAX.**
 - R2: La recherche peut être stoppée sous tout nœud MAX ayant une valeur Alpha supérieure ou égale à la valeur Beta de n'importe lequel de ses ancêtres MIN.**
 - Alpha-Beta exprime une relation entre les nœuds de profondeur n et ceux de profondeur $n+2$ pour lesquels les sous-arbres à la profondeur $n+1$ peuvent être élagués.
 - Dans le pire cas, Alpha-Beta n'élague aucun sous-arbre et examine autant de nœuds que Minimax.

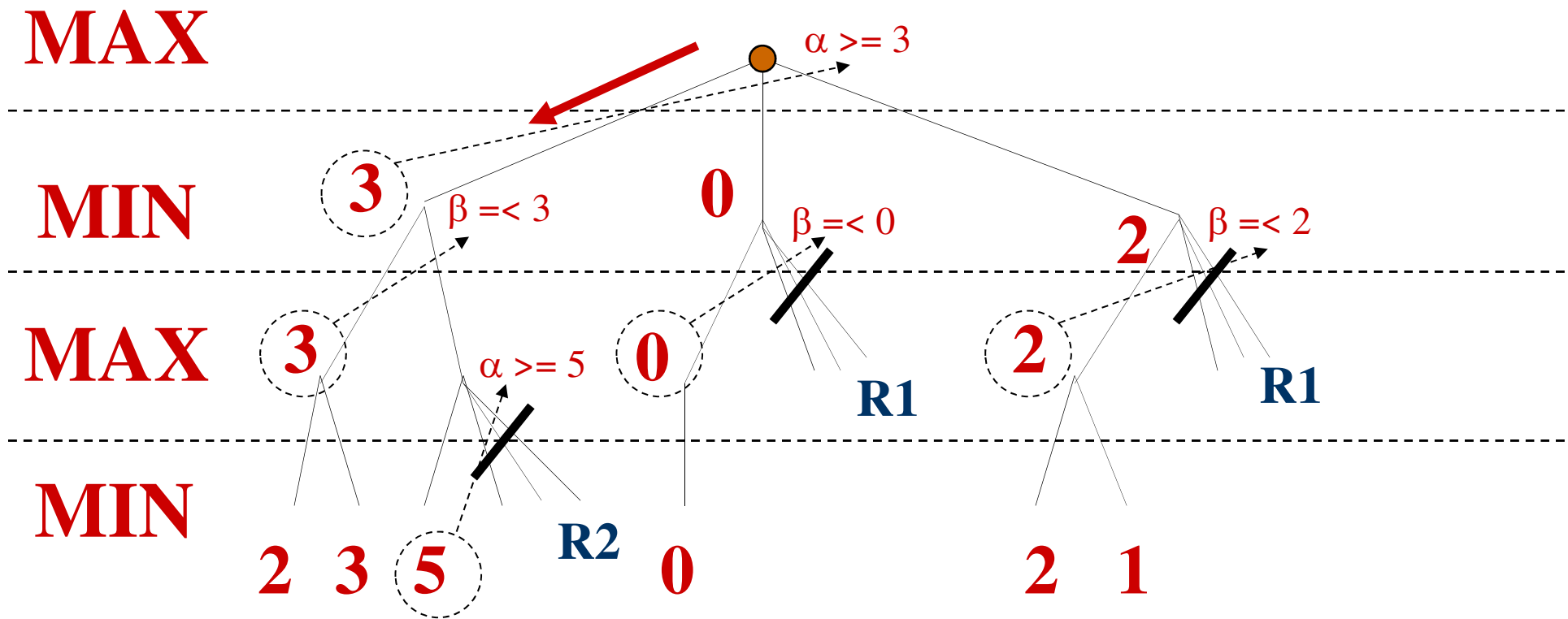
Recherche heuristique pour les jeux

- Procédure Alpha-Beta
 - Exemple



Recherche heuristique pour les jeux

- Procédure Alpha-Beta
 - Exemple



Recherche heuristique pour les jeux

- Lorsqu'on utilise la procédure Minimax avec ou sans élagage, le joueur Max est notre joueur, qu'on veut faire gagner.
- Pour certains jeux, il est préférable de laisser son adversaire jouer en premier comme on l'a vu avec le jeu de Nim.

Conclusion

- Les jeux sont une application importante de l'IA :
 - d'un point de vue historique
 - actuellement avec les jeux vidéos
- Les techniques étudiées sont efficaces et bien connues, donc encore utilisées
- Il existe aussi d'autres techniques de recherche intelligente ou variantes de techniques