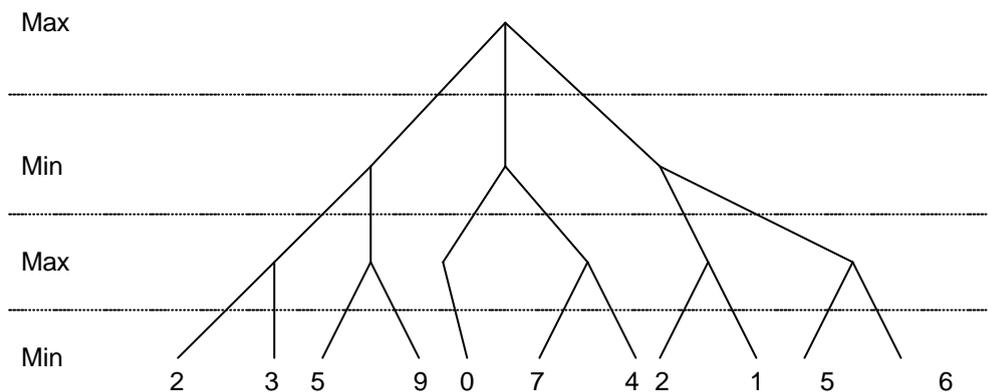


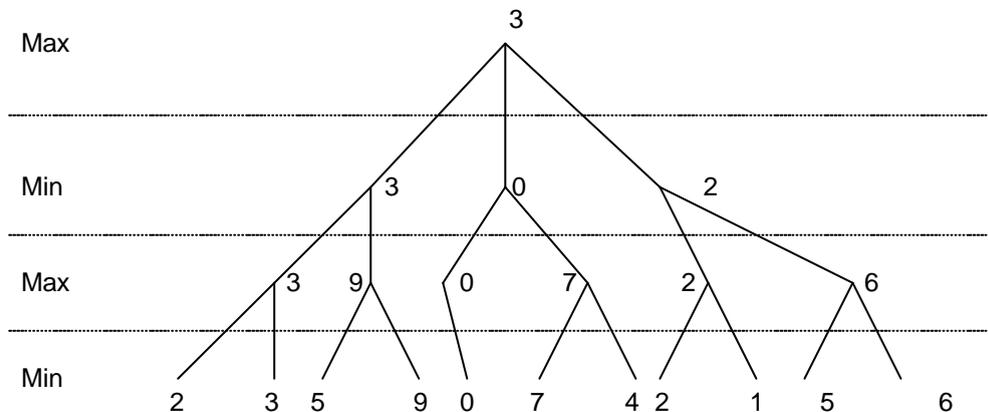
Minimax avec un niveau fixe

On ne peut pas toujours développer l'arbre de recherche à cause du problème d'explosion combinatoire. Dans ce cas, on va développer l'arbre jusqu'à un certain niveau.

Supposons qu'on obtienne l'arbre (développé par le programme) suivant avec les valeurs heuristiques données aux feuilles.



Ensuite, le programme fait remonter les valeurs Min (enfants) ou Max (enfants) selon le cas.



Max choisira ensuite le meilleur coût maximisant ainsi ses gains.

Le Minimax est un algorithme à 2 passes :

- 1) on développe l'arbre de recherche jusqu'à un certain niveau et on évalue la fonction heuristique pour chaque feuille ;
- 2) on propage les valeurs jusqu'à la racine.

Alpha-Béta (a-b)

C'est un algorithme plus intelligent que Minimax, qui va permettre d'élaguer l'arbre de recherche du Minimax.

Plutôt que de chercher tout l'espace jusqu'à la profondeur désirée, la procédure α - β procède en profondeur d'abord. Deux valeurs, notées α et β , sont créées pendant la recherche :

- valeur α , associée aux nœuds Max, ne peut jamais diminuer ;
- valeur β , associée aux nœuds Min, ne peut jamais augmenter ;

Exemple :

Si la valeur de α est égale à 6, alors Max n'a pas à considérer les nœuds avec des valeurs ≤ 6 . La valeur α est donc la pire valeur que peut obtenir Max.

Si Min a une valeur β égale à 6, il n'a pas à considérer les nœuds Max qui ont une valeur ≥ 6 .

On démarre la recherche α - β , en suivant la profondeur d'abord et on applique l'évaluation heuristique à un état et ses enfants. Les étapes suivies sont numérotées de [1] à [15].

