



# Résolution de problèmes

---

TP des missionnaires et canibales

AJALE Saâd  
BENHADDOU Lahcen

20/03/2024

Ce rapport présente une modélisation pratique du tp du missionnaires et cannibales ,ainsi qu'une analyse comparative tout au long le projet

## TABLES DE MATIERES :

I.	INTRODUCTION.....	3
II.	CONCEPTION.....	4
III.	MODELISATION.....	5
1.	RECHERCHE EN LARGEUR.....	6
2.	RECHERCHE EN PROFENDEUR.....	8
IV.	COMPARAISON.....	10
V.	CONCLUSION.....	12

## I. INTRODUCTION

Le défi des missionnaires et des cannibales suscite une réflexion intrigante, mettant en lumière la nécessité d'une navigation prudente pour garantir la sécurité des missionnaires lors de la traversée du fleuve. Ce problème offre une opportunité fascinante d'explorer la modélisation complexe des interactions entre les individus et la barque, démontrant ainsi une approche stratégique pour résoudre ce casse-tête délicat.

## II. CONCEPTION

Le programme vise à déterminer le chemin entre l'état initial et l'état final, en tenant compte des contraintes liées au nombre de missionnaires et de cannibales sur chaque rive. Nous avons introduit une structure de données triple pour représenter l'état actuel, incluant le nombre de cannibales, de missionnaires et la position de la barque. Ensuite, nous avons développé un ensemble de règles pour déplacer les individus en toute sécurité, en limitant le nombre de cannibales présents sur une rive à tout moment. En outre, une fonctionnalité a été ajoutée pour permettre à l'utilisateur de saisir l'état final, ce qui détermine l'objectif de la traversée.

### III. MODELISATION

La modélisation du problème des missionnaires et des cannibales consiste à définir un ensemble d'états possibles, des règles de transition et un état but à atteindre.

#### Espace d'États:

L'espace d'états est défini par des triplets comprenant le nombre de cannibales restants sur la rive, le nombre de missionnaires restants sur la rive final, et la position de la barque. La position de la barque est représentée par 0 pour la côte initiale et 1 pour la côte finale. Les nombres de cannibales et de missionnaires sont compris entre 0 et 3, et la position de la barque peut être soit 0 (côte initiale) soit 1 (côte finale).

#### États Initial et But:

L'état initial du système se caractérise par la présence des trois missionnaires et des trois cannibales sur la côte initiale, avec la barque également sur cette même côte. Il est représenté par le triplet (nombre de cannibales, nombre de missionnaires, position de la barque) égal à (0, 0, 0). L'état but est saisi par l'utilisateur.

#### Règles:

Deux individus sont transportés dans la barque au maximum parmi les 3 missionnaires et les 3 cannibales :

Règle 1 : 2 cannibale , 0 missionnaire

Règle 2 : 0 cannibale , 2 missionnaire

Règle 3 : 1 cannibale , 1 missionnaire

Règle 4 : 0 cannibale , 1 missionnaire

Règle 5 : 1 cannibale , 0 missionnaire

#### Contrainte :

Pour assurer la sécurité des missionnaires, les contraintes suivantes sont imposées, que ce soit sur la rive initiale ou la rive finale : Le nombre de missionnaires doit être égal à zéro ou supérieur au nombre de cannibales présents sur la même rive.

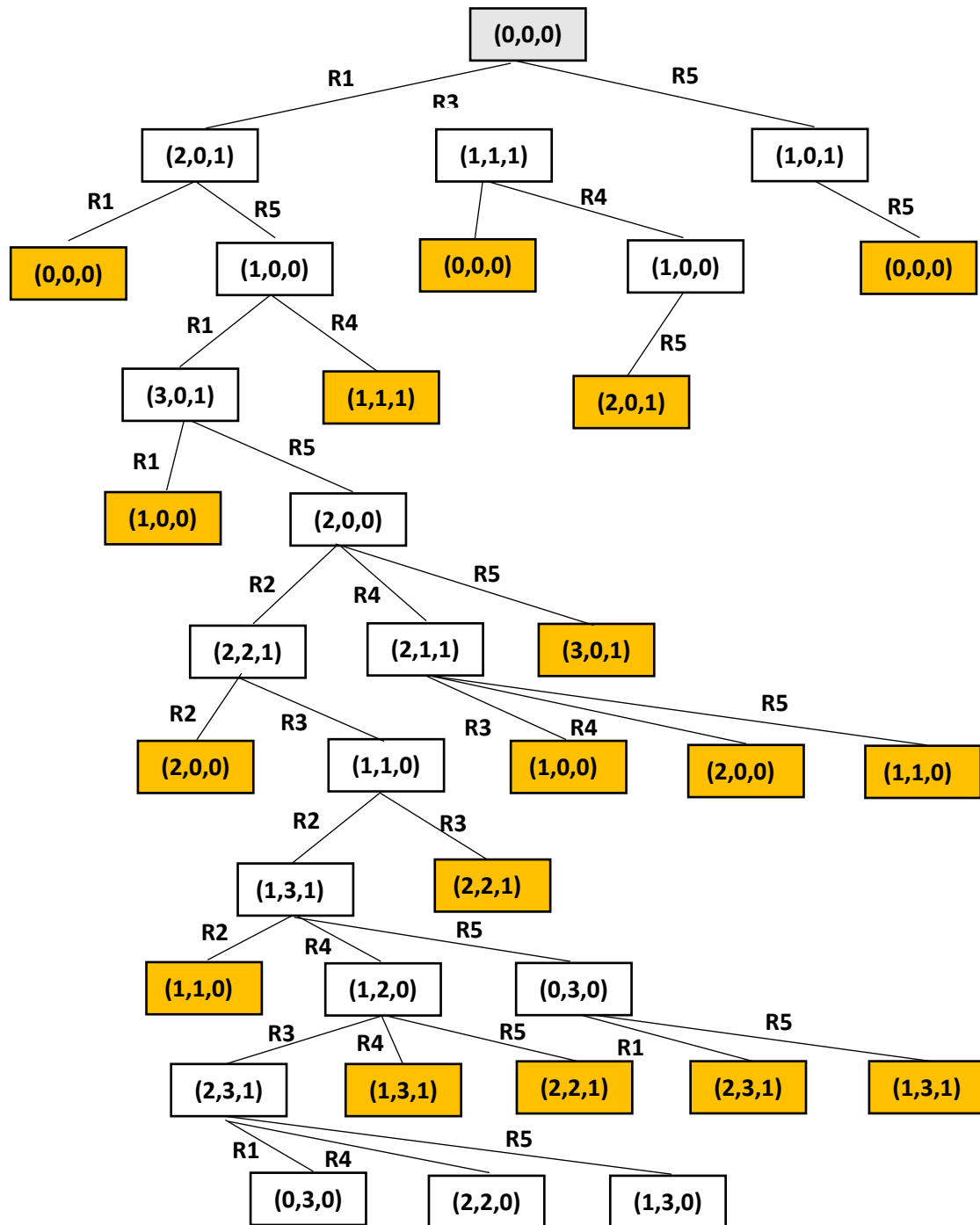
#### Objectif de la Modélisation:

Cette modélisation vise à déterminer le chemin qui mène à l'état but.

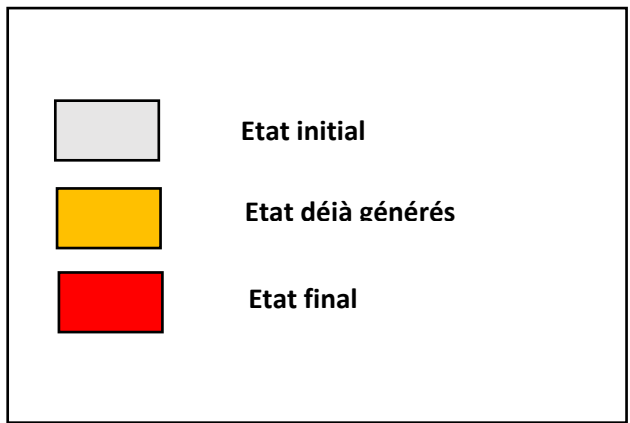
## **1. RECHERCHE EN LARGEUR**

En ce qui concerne la recherche en largeur, les nœuds générés sont stockés dans une file, suivant ainsi une approche systématique pour explorer les états potentiels dans un ordre étendu avant de passer aux niveaux suivants.

Voici maintenant un schéma de résolution en respectant l'ordre des règles énoncées initialement, avec pour objectif d'atteindre l'état final spécifié, soit **(3,3,1)**.



(3,3,1)

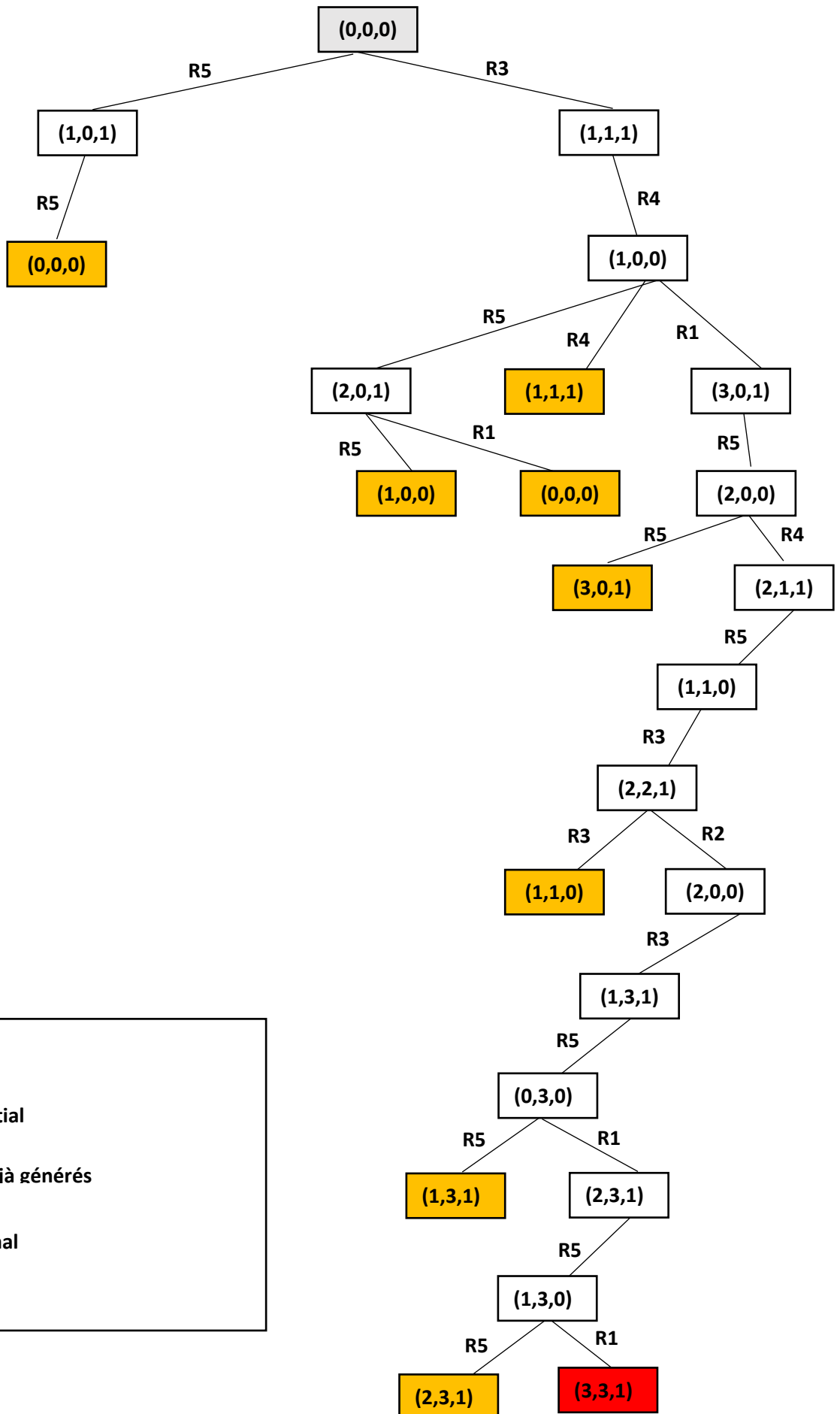


## 2. RECHERCHE EN PROFONDEUR

Quant à la recherche en profondeur, elle privilégie une approche plus ciblée en explorant autant que possible dans une branche avant de revenir en arrière, utilisant une pile pour stocker les nœuds et descendant en profondeur dans l'arbre d'états.

De même nous avons réalisé un schéma de résolution en respectant l'ordre des règles énoncées initialement, avec pour objectif d'atteindre l'état final spécifié, soit (3,3,1).





## IV. COMPARAISON

Nous avons opté pour une conception de la structure 'état' basée sur l'utilisation de bits, puisque nos données nécessitent seulement 5 bits au lieu de 3 entiers. Cette approche nous a permis d'économiser de l'espace mémoire, ce qui est particulièrement bénéfique étant donné que nous utilisons des structures arborescentes qui consomment beaucoup de mémoire.

### Utilisation des entiers

```
typedef struct {  
    int ncanibal; //a la cote final  
    int nmission;  
    int barque; //1 existe a la cote finale 0 si non  
} etat;
```

### Mémoire occupée pour une variable etat :

```
etat est de size :12  
Process returned 0 (0x0)   execution
```

### Utilisation des bits :

```
#define FONCTIONS_H  
  
typedef struct {  
    unsigned int ncanibal : 2; //a la cote final  
    unsigned int nmission : 2;  
    unsigned int barque : 1; //1 existe a la cote finale 0 si non  
} etat;
```

### Mémoire occupée pour une variable etat :

```
etat est de size :4  
Process returned 0 (0x0)   execu  
Press any key to continue.  
_
```

Donc pour trouver l'état final (3,3,1) à partir de l'état initial (0,0,0) on stocke au moins 60 états pour la recherche en largeur et 36 états pour la recherche en profondeur :

```
Menu:
1. Recherche en largeur
2. Recherche en profondeur
3. Quitter
Votre choix : 1

Succes ! arret sur le noeud :(3,3,1)
Nombre de noeuds explores :36
le chemin pour avoir 3 canibal ,3 missionnaire a la cote finale est comme suit :
-(3,3,1)--(2,2,0)--(2,3,1)--(1,2,0)--(1,3,1)--(1,1,0)--(2,2,1)--(2,0,0)--(3,0,1)--(1,0,0)--(2,0,1)--(0,0,0)-

Menu:
1. Recherche en largeur
2. Recherche en profondeur
3. Quitter
Votre choix : 2

Succes ! arret sur le noeud :(3,3,1)
Nombre de noeuds explores :60
le chemin pour avoir 3 canibal ,3 missionnaire a la cote finale est comme suit :
-(3,3,1)--(1,3,0)--(2,3,1)--(0,3,0)--(1,3,1)--(1,1,0)--(2,1,1)--(2,0,0)--(3,0,1)--(1,0,0)--(1,1,1)--(0,0,0)-
```

### Mémoire occupée pour Utilisation des entiers :

#### Recherche en largeur :

36\*12=432 (octets)

#### Recherche en profondeur :

60\*12=720 (octets)

### Mémoire occupée pour Utilisation des bits :

#### Recherche en largeur :

36\*4=144 (octets) conservation du 288 octets

#### Recherche en profondeur :

60\*4=240 (octets) conservation du 480 octets

## **VI. CONCLUSION**

En conclusion, ce projet sur les missionnaires et cannibales a été une expérience stimulante qui nous a permis d'appliquer divers concepts de modélisation, de comparaison et de conception. Nous avons réussi à relever les défis liés à la manipulation et à l'optimisation de l'utilisation de la mémoire, en introduisant des solutions innovantes qui conservent davantage de mémoire. Cette expérience nous a également permis d'approfondir notre compréhension des différentes approches de conception de recherche. En résumé, ce projet nous a fourni des connaissances précieuses et des compétences pratiques qui seront indéniablement bénéfiques pour nos projets futurs et notre développement professionnel.