 Guillaume DUBUISSON DUPLESSIS

Pauline REQUENA

Arnaud FAURE

Département GM - 4ème année

**Rapport de mini-projet C++**

**Sujet : Résolution du monde des cubes par éco-résolution**

****

**A l’attention de Mme Jean-Philippe Kotowicz**

Année scolaire 2008/2009 – 2ème semestre GM4

**Sommaire**

**Introduction** . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4

**I/ Etude de l’éco-résolution**

1. Principe . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6
2. Le monde des cubes . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8

**II/ Modélisation d’un problème par l’éco-résolution : Les tours de Hanoi**

1. Réification . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 12
2. Diagramme des classes . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 15

**III/ Implémentation du problème en langage Java**

1. Description des classes . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 26
2. Langage algorithmique . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 19
3. Langage Java . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 20
4. Exécution du programme. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 26
5. Exécution 1 : ligne de commande . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 19
6. Exécution 2 : interface graphique . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 20

**IV/ Déroulement de la méthode sur un exemple**

**Conclusion** . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 64

**Webographie** . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 65

**Introduction**

blabla

**Partie 1 : Etude de l’éco-résolution**

1. **Principe de l’éco-résolution**

L’éco-résolution est utilisée pour la résolution des problèmes en Intelligence Artificielle.

Elle se compose de 2 parties :

* Un protocole suivi par l’ensemble des agents, qui est un noyau indépendant du problème à résoudre.
* Un code de comportements des éco-agents spécifiques au domaine.

1. **Caractéristiques d’un éco-agent**

Un éco-agent est en quête perpétuelle d’un état de satisfaction. Il peut être gêné par d’autres éco-agents et, à ce moment, il agresse les gêneurs qui fuient.

Un éco-agent est caractérisé par :

* **Un but** : il s’agit d’un autre éco-agent avec lequel il dit être en relation de satisfaction.
* **Un état interne** : satisfait, en recherche de satisfaction, en fuite ou en recherche de fuite.
* **Des actions élémentaires** : Elles dépendent du domaine et correspondent aux comportements de satisfaction ou de fuite.
* **La perception des gêneurs :** Il s’agit de la perception des éco-agents qui empêchent l’éco-agent courant d’être satisfait ou de fuir.
* **Des dépendances :** les éco-agents dont l’éco-agent courant est le but. Elles sont satisfaites uniquement si cet éco-agent est satisfait.

1. **Comportement d’un éco-agent**

Un éco-agent a la volonté d’être satisfait. Il cherche à se trouver dans un état de satisfaction. S’il est empêché par des gêneurs alors il les agresse.

Un éco-agent a l’obligation de fuir. Si un éco-agent est agressé, il doit trouver une place ou fuir.

1. **Représentation d’un éco-agent et diagramme d’état**

Un éco-agent peut être représenté par un automate à états-finis. Les différents états de l’automate sont les suivants :

**S** satisfait

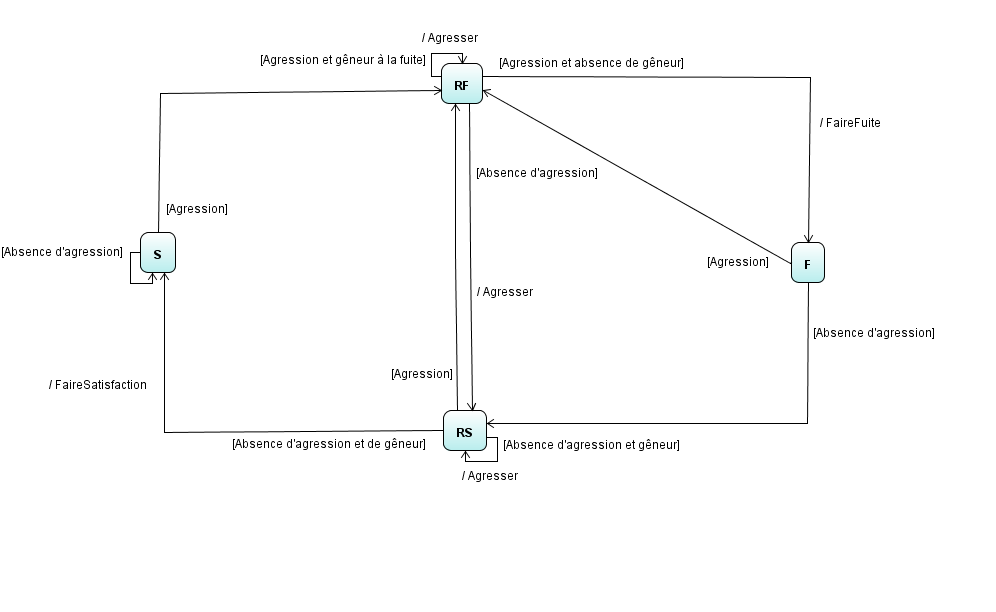
**RS** recherche de satisfaction

**F**  fuite

**RF** recherche de fuite

Un éco-agent peut effectuer 3 actions différentes :

1. Agresser(EcoAgent e)
2. FaireSatisfaction
3. FaireFuite



**Diagramme d’états d’un EcoAgent**

* **EcoAgent satisfait**

**Etat initial :**

L’EcoAgent est dans son état de satisfaction.

**Différents scénarios possibles :**

Si l’EcoAgent n’est pas agressé, il reste dans l’état « satisfait ».

Si l’EcoAgent est agressé par un autre EcoAgent, il passe dans l’état « en recherche de fuite ».

* **EcoAgent en recherche de satisfaction**

**Etat initial :**

L’EcoAgent n’est pas agressé mais il n’est pas dans son état de satisfaction. Il cherche donc à l’atteindre.

**Différents scénarios possibles :**

Si l’EcoAgent n’est pas agressé et qu’aucun autre EcoAgent ne le gêne pour atteindre sa satisfaction, alors il se satisfait avec la méthode FaireSatisfaction et entre dans l’état « satisfait ».

Si l’EcoAgent n’est pas agressé mais qu’au autre EcoAgent le gêne pour atteindre son état de satisfaction, il agresse le gêneur avec la méthode Agresser pour l’obliger à fuir, et retourne dans un état « en recherche de satisfaction ».

Si l’EcoAgent est agressé par un autre EcoAgent, il passe dans l’état « en recherche de fuite ».

* **EcoAgent en recherche de fuite**

**Etat initial :**

L’EcoAgent a été agressé. Il est donc en recherche de fuite.

**Différents scénarios possibles :**

Si aucun gêneur n’empêche l’EcoAgent de fuir, il passe dans l’état « fuite » avec la méthode FaireFuite.

Si l’EcoAgent ne peut pas fuir à cause d’un gêneur, il agresse ce dernier avec la méthode « Agresser », afin de l’obliger à fuir, et retourne dans un état « en recherche de fuite ».

Si l’EcoAgent n’est plus agressé, il passe à l’état « en recherche de satisfaction ».

* **EcoAgent en fuite**

**Etat initial :**

L’EcoAgent a fui suite à une agression.

**Différents scénarios possibles :**

Si l’EcoAgent est de nouveau agressé par un autre EcoAgent, il retourne dans l’état « en recherche de fuite ».

Si aucun EcoAgent n’agresse cet EcoAgent, il peut repasser à l’état « en recherche de satisfaction ».

1. **Un exemple d’utilisation : le monde des cubes**

**Présentation**

Le problème des tours de Hanoï est un [jeu de réflexion](http://fr.wikipedia.org/wiki/Jeu_de_r%C3%A9flexion) imaginé par le [mathématicien](http://fr.wikipedia.org/wiki/Math%C3%A9maticien) français [Édouard Lucas](http://fr.wikipedia.org/wiki/%C3%89douard_Lucas).

Il consiste à déplacer des disques de diamètres différents d'une tour de « départ » à une tour d'« arrivée » en passant par une tour « intermédiaire » et ceci en un minimum de coups, tout en respectant les règles suivantes :

* on ne peut déplacer plus d'un disque à la fois,
* on ne peut placer un disque que sur un autre disque plus grand que lui ou sur un emplacement vide.

On suppose que cette dernière règle est également respectée dans la configuration de départ.



**Modélisation du système d’éco-résolution**

1. **Les éco-agents**

Les disques numérotés de 1 à n par taille croissante

* D1 correspondant au plus petit disque
* Dn correspondant au plus grand disque

Les 3 supports correspondants aux 3 tiges

* S1 pour la tige de départ
* S2 pour la tige intermédiaire
* S3 pour la tige finale

Ces 3 derniers agents sont toujours satisfaits car ils sont satisfait à l’état initial et ne peuvent pas être agressés.

1. **Conditions de satisfaction**

Considérons le cas où n=5, le système se trouve au départ dans l’état initial suivant :

{ sur(D1,D2), sur(D2,D3), sur(D3,D4), sur(D4,D5), sur(D5,S1) }

Le but est de se trouver dans l’état final suivant :

{ sur(D1,D2), sur(D2,D3), sur(D3,D4), sur(D4,D5), sur(D5,S3) }

1. **Relation de dépendances**

{ dépendDe(D1,D2), dépendDe(D2,D3), dépendDe(D3,D4), dépendDe(D4,D5), dépendDe(D5,S3) }

1. **Contraintes de fuite**

Lorsqu’il est agressé, un éco-agent a donc au maximum 2 lieux de fuite possibles, correspondant aux 2 autres tours.

Cependant, les lieux de fuite possibles peuvent être réduits à 1 ou 0 emplacement à cause des contraintes suivantes :

* Un disque peut se poser sur un support vide.
* Si la tour n’est pas vide, un disque ne peut se poser que sur un disque plus grand que lui.

**Stratégie de résolution**

A chaque étape du jeu, on définit une tour initiale, une tour intermédiaire et une tour finale.

Considérons une tour initiale de n disques. Au début du jeu, la tour initiale est la tour n°1, la tour intermédiaire est la tour n°2, et la tour finale est la tour n°3.

La stratégie de résolution consiste à placer la tour supérieure de taille (n-1) disques sur la tour intermédiaire, afin de dégager le disque le plus large et de le placer sur la tour finale.

Une fois cette première étape effectuée, la tour initiale devient la tour n°2 et la tour intermédiaire devient la tour n°1. On réapplique ainsi la même stratégie sur la tour initiale, etc… jusqu’à atteindre l’état final.

***Exemple***