Le jeu de Juniper-Green

Boris LAVAL - Olivier SICARD

6 novembre 2017

IREM DE LA REUNION

Table des matières

I Introduction **3**

II Les règles du jeu **3**

III Premières réflexions mathématiques **3**

IV Le combat des IA **4**

V Stratégie gagnante pour le jeu de Juniper-Green **5**
V.1 Mise en place des éléments du jeu 5
V.2 Construction de la stratégie gagnante 6

VI Exemples de stratégies gagnantes dans deux cas simples : n=6 et n=8 **7**
VI.1 Une stratégie gagnante pour le joueur 2 lorsque n=6 7
VI.2 Une stratégie gagnante pour le joueur 1 lorsque n=8 8
VI.3 Récapitulatif des stratégies gagnantes 9

VII Algorithme de recherche de stratégie gagnante **10**

VIII Conclusion **10**

IX Annexes **11**

I Introduction

Le jeu de Juniper Green, aussi appelé ”jeu des multiples et des diviseurs” est un jeu mathématique opposant deux joueurs. Il a été créé par Richard Porteous, enseignant à l'école de Juniper Green, auquel il doit son nom.

Nous avons découvert l'existence de ce jeu un peu par hasard dans un sujet d'examen de BTS SIO, et dès lors il est devenu pour nous et pendant plusieurs mois l'objet de plusieurs défis et de grandes discussions. Le fil conducteur de nos discussions était principalement : ”Comment fait-on pour gagner au jeu de Juniper Green ?” Cet article est le compte rendu de nos recherches.

Nous tenons à remercier Alexandre MANSARD de nous avoir parlé de la détermination positionnelle des jeux d'accessibilité sans quoi la démonstration d'existence de stratégies gagnantes aurait été impossible.

II Les règles du jeu

Commençons par détailler les règles du jeu. Elles sont plutôt simples :

— Règle 1 : Considérons N un entier naturel non nul, le jeu de Juniper Green se joue avec les nombres entiers de 1 à N. A l'origine du jeu il semblerait que N valait 20.
— Règle 2 : Le premier joueur choisi un nombre pair entre 1 et N.
— Règle 3 : Une fois qu'un nombre a été choisi il ne pourra plus jamais être utilisé.
— Règle 4 : Chaque joueur doit choisir un nombre qui est soit un diviseur soit un multiple du nombre précédent.
— Règle 5 : Le joueur qui ne peut plus jouer perd la partie.

III Premières réflexions mathématiques

Concentrons nous sur le jeu de Juniper Green pour $N = 20$, et commençons par présenter le tableau des coups possibles.

LISTE DES COUPS POSSIBLES

coup précédent	coups possibles																			
1		2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
2	1			4		6		8		10		12		14		16		18		20
3	1					6			9			12			15			18		
4	1	2						8				12				16				20
5	1									10					15					20
6	1	2	3									12						18		
7	1													14						
8	1	2		4												16				
9	1		3															18		
10	1	2			5															20
11	1																			
12	1	2	3	4		6														
13	1																			
14	1	2					7													
15	1		3		5															
16	1	2		4				8												
17	1																			
18	1	2	3			6			9											
19	1																			
20	1	2		4	5					10										

Pour i allant de 1 à 20, les entiers de la ligne i sont les coups que peut jouer le joueur lorsque son adversaire vient de jouer le nombre i. Il ne faudra pas oublier de retirer de cette liste les coups déjà joués précédemment pour ne pas déroger à la règle 3.

Ce premier tableau montre le déséquilibre en terme de nombre de coups possibles suivant ce qui a été joué précédemment. Prenons l'exemple du premier coup d'une partie :

— Jouer le nombre 2 donne à son adversaire 10 coups différents possibles

— Jouer 14 n'offre que 3 possibilités à son adversaire

Le tableau ci-dessous montre le nombre de coups possibles suivant le coup précédent rangé dans l'ordre croissant des coups possibles :

coup précédent	11-13-17-19	7	9-14-15	5-8-10-16	6-12-18-20	3-4	2	1
nb coups possibles	1	2	3	4	5	6	10	19

Regardons ce nouveau tableau de plus près :

Commençons par la première colonne ; elle contient les nombres 11 ; 13 ; 17 et 19. Ce sont tous les nombres premiers compris entre $\frac{N}{2}$ et N. (ici 10 et 20)

Chacun de ces nombres étant supérieur à $\frac{N}{2}$ ils n'ont aucun multiple plus petit que N, et étant premier ils n'ont aucun diviseur (autre que 1). L'adversaire est alors obligé de jouer 1 s'il est encore disponible, sinon il a perdu !

La suite logique est que celui qui joue 1 perd, car alors son adversaire jouera le nombre qu'il voudra parmi 11-13-17-19 et notre joueur n'aura plus de coup à jouer. le but du jeu est donc d'obliger notre adversaire à jouer 1.

Regardons maintenant la deuxième colonne : Théoriquement 7 a deux successeurs qui sont 1 et 14. Dans la réalité d'une partie ce n'est pas tout à fait le cas. En effet la règle 2 oblige le premier joueur à jouer un nombre pair (on comprend cette règle maintenant, car si ce n'était pas le cas le premier joueur commencerait par 11 obligeant son adversaire à jouer 1 ce qui assurerait la victoire du premier joueur comme nous l'avons vue juste au dessus.)

Donc 7, qui n'est pas pair, est forcément le successeur d'un entier joué précédemment, et celui-ci ne peut être que 1 ou 14. Or si le coup précédent est 1 personne n'aurait l'idée d'aller jouer 7 : il préfèrerait jouer par exemple 11 et gagner la partie. On peut donc en déduire que le prédécesseur de 7 est 14. Après le 7 il ne reste donc que le 1 à jouer, ce qui clôture la partie au tour suivant.

En conclusion celui qui joue 14 perd car l'autre joue 7 l'obligeant à jouer 1.

Pour ce qui est des autres colonnes le nombre de cas possibles augmente de façon exponentielle et il n'est plus possible de raisonner comme nous venons de le faire. Heureusement d'ailleurs car alors le jeu aurait peu d'intérêt, mais dans ces conditions comment faire pour gagner à ce jeu ?

IV Le combat des IA

Notre but était de comprendre comment gagner à ce jeu. Nous n'avions pas de stratégie gagnante, nous ne savions même pas s'il y en existait une ! Dans ces conditions comment jouer au mieux ?

L'idée du combat entre Intelligences Artificielles nous est venu assez naturellement. Deux camps s'affrontaient car basés sur deux concepts différents :

— Une famille d'IA étaient de type "humaine". les IA humaines maitrisaient les concepts mathématiques détaillés dans la section précédente. Par exemple aucune IA humaine n'aurait joué 14 ! L'IA humaine la plus aboutie avait pour stratégie gagnante de jouer le coup offrant le moins de possibilité à son adversaire au coup suivant.

— Une dernière IA était de type "inhumaine" avec une approche probabiliste, c'est à dire qu'elle utilisait toute la puissance de la machine pour parcourir les voies et calculer pour chaque coup qu'elle pouvait jouer la probabilité qu'elle avait de gagner. Elle jouait alors le coup qui lui donnait le plus de chances de gagner au final.

Voici le tableau récapitulatif de nos matchs :

		JOUEUR 2				
	taux de win du joueur 1 pour nbmax=20	IA0 - full random	IA1 - humaine	IA2 - semi expert humain	IA 3	IA4 inhumaine
JOUEUR 1	IA0	47,70%	47,70%	41,70%	52,60%	
	IA1	47,70%	44,60%	41,70%	52,40%	
	IA2	55,10%	55,10%	51,70%	61,40%	
	IA3	90,20%	90,20%	90,20%	100%	80%
	IA4				80%	66%

Nous avons d'abord fait jouer entre elles nos IA0-IA1-IA2-IA3 qui sont toutes des IA humaines de plus en plus abouties. Les résultats du tableau nous incitaient à conclure que l'IA3 était clairement l'IA la plus experte des IA humaines, mais pour qu'elle exprime vraiment son potentiel encore fallait-il que l'IA3 commence la partie.

Nous avons ensuite fait jouer l'IA3 (experte humaine) contre l'IA4 (inhumaine).Le résultat était plutôt surprenant : L'IA3 gagnait contre l'IA4 huit fois sur dix quand elle commençait, et l'IA4 gagnait aussi huit fois sur dix contre l'IA3 quand elle commençait.

Quand on est face à ce genre de résultat, on ne peut conclure qu'une seule chose : " il existe surement une stratégie gagnante pour le joueur1 au jeu de Juniper Green lorsque N=20 !"

V Stratégie gagnante pour le jeu de Juniper-Green

V.1 Mise en place des éléments du jeu

Considérons l'alphabet $\mathcal{A} = [\![1, n]\!]$ formé des entiers naturels de 1 à n, ainsi que $\mathcal{V}$ l'ensemble des mots sur cet alphabet avec l'orthographe suivante :

— la première lettre du mot est paire

— les lettres suivantes sont des multiples ou des diviseurs de la lettre précédente

— toutes les lettres sont différentes

On peut alors construire le graphe orienté associé au jeu de Juniper-Green de la manière suivante :

— $\mathcal{V}$ est l'ensemble des sommets du graphe

— On définit les arrêtes du jeu par une relation binaire sur l'ensemble des sommets de sorte que deux sommets m_1 et m_2 soient reliés ($m_1 \to m_2$) si et seulement si le mot m_2 privé de sa dernière lettre est égal à m_1
— Notons $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ l'ensemble de toutes les arrêtes du jeu

Les mots à une lettre paire constituent les sommets de départ du jeu, on les notera $\mathcal{D}$.
Soit $m \in \mathcal{V}$ nous appellerons les successeurs de m l'ensemble $m\mathcal{E} = \{m' \in \mathcal{V}, (m, m') \in \mathcal{E}\}$

Une partie se déroule ainsi :
— Le joueur 1 (J_1) positionne un pion sur un sommet de départ.
— Le joueur 2 (J_2) fait avancer le pion sur un sommet adjacent et ainsi de suite
— J_1 gagne si J_2 ne peut plus avancer (et inversement)

Remarquons que J_1 avance son pion sur des mots de longueur impaire alors que J_2 les avance sur des mots de longueur paire.

V.2 Construction de la stratégie gagnante

Théorème :
Le jeu de Juniper-Green admet une stratégie gagnante pour l'un des joueurs.

Preuve :
Essayons de construire une stratégie gagnante pour J_1.
Pour que J_1 gagne, il faut que J_2 pose le pion sur un mot de longueur paire et finissant par 1. En effet, cela signifie que J_2 vient de jouer 1 et nous savons dans ce cas qu'il suffit de jouer un nombre premier plus grand que $\frac{n}{2}$ pour remporter la partie.

Notre but est de fabriquer un piège P en forme de strates P_i obligeant J_2 à jouer le nombre 1. Pour cela notons P_0 l'ensemble de tous les mots de longueur paire et finissant par 1 : c'est le coeur de notre piège.

Considérons $P_1 = \{m \in \mathcal{V}, m\mathcal{E} \subseteq P_0\}$, c'est à dire l'ensemble des mots de longueur impaire ayant tous leurs successeurs dans P_0. C'est la première strate de notre piège.
Si J_1 pousse son pion dans P_1, alors J_2 sera obligé de jouer dans P_0 et J_1 gagne.

Considérons ensuite $P_2 = \{m \in \mathcal{V}, \exists m' \in P_1 \cap m\mathcal{E}\}$ c'est à dire l'ensemble de tous les mots de longueur paire pour lesquels au moins un successeur appartient à P_1. Quand J_2 pousse son pion dans P_2 alors la stratégie de J_1 consiste à jouer dans P_1.

Puis :
$P_3 = \{m \in \mathcal{V} \setminus P_1, m\mathcal{E} \subseteq P_2 \cup P_0\}$
$P_4 = \{m \in \mathcal{V} \setminus P_2, \exists m' \in (P_3 \cup P_1) \cap m\mathcal{E}\}$
$P_5 = \{m \in \mathcal{V} \setminus (P_1 \cup P_3), m\mathcal{E} \subseteq P_4 \cup P_2 \cup P_0\}$

On réitère alors les étapes précédentes de sorte que :
— P_{2k+1} est l'ensemble des mots de longueur impaire non encore contenu dans le piège et ayant tous leurs successeurs dans $\cup_{i=0}^{k} P_{2k}$
— P_{2k} est l'ensemble de tous les mots de longueur paire non encore contenu dans le piège et pour lesquels au moins un successeur appartient à $\cup_{i=0}^{k} P_{2i-1}$

Considérons maintenant la suite (V_n) où $V_n = \cup_{i=0}^{n} P_i$. (V_n) est une suite croissante de parties de $\mathcal{V}$ qui est un ensemble fini, elle est donc convergente. Notons P sa limite.

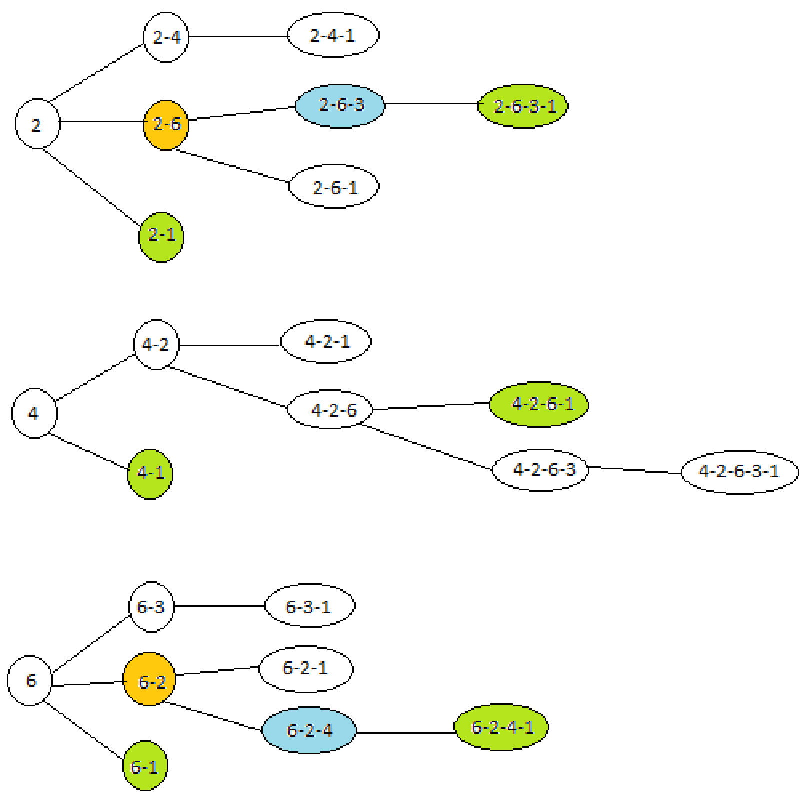
Si au moins un sommet de $\mathcal{D}$ appartient à P alors J_1 possède une stratégie gagnante. Dans le cas contraire $D \subseteq \overline{P}$ et dans ce cas c'est J_2 qui possède une stratégie gagnante. Dans les deux cas il existe bien une stratégie gagnante pour l'un des joueurs.
cqdf.

VI Exemples de stratégies gagnantes dans deux cas simples : n=6 et n=8

Pour les valeurs de n petites il est assez aisé de dessiner les arbres à la main et de colorier le paradis du joueur 1. Nous allons le faire pour n=6 puis pour n=8.

VI.1 Une stratégie gagnante pour le joueur 2 lorsque n=6

Posons $\mathcal{A} = \{1, 2, 3, 4, 5, 6\}$ notre alphabet, l'ensemble des mots de départ est alors $\mathcal{D} = \{2, 4, 6\}$. Voici l'arbre complet du jeu sur lequel nous avons colorié le piège de J_1 :



- Nous avons colorié en vert le cœur du piège, c'est à dire les mots de longueur paire et finissant par 1 : $P_0 = \{21, 41, 61, 2631, 4261, 6241\}$
- Nous avons colorié en bleu la strate P_1, c'est à dire l'ensemble des mots de longueur impaire dont tous leurs successeurs sont dans P_0. Par exemple le mot "263" est

dans P_1 car "263" possède un unique successeur qui est dans P_0, mais le mot "426" n'appartient pas à P_1 car "426" possède deux successeurs dont un des deux ne fait pas parti de P_0. Au final $P_1 = \{263, 624\}$

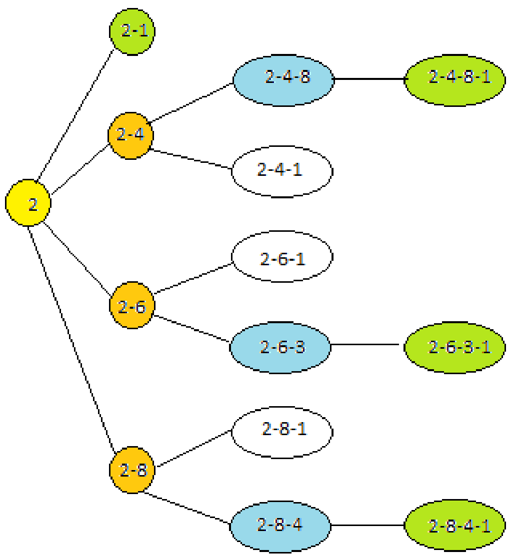
— Nous avons colorié en orange la strate P_2. Ce sont les mots de longueur paire non encore coloriés possédant au moins un successeur dans P_1. Nous obtenons $P_2 = \{26, 62\}$

— La strate P_3 doit être constituée des mots de longueur impaire non encore coloriés et possédant tous leurs successeurs dans $P_0 \cup P_2$. Or aucun des mots de longueur impaire non coloriés restants ne vérifient cette propriété donc $P_3 = \emptyset$

Au final $P = P_0 \cup P_1 \cup P_2$ c'est à dire l'ensemble de tous les sommets coloriés, et puisque $\mathcal{D} \subseteq \overline{P}$, J_2 a une stratégie gagnante qui consiste à toujours jouer les positions non coloriées : $\overline{P}$ est le piège de J_2.

VI.2 Une stratégie gagnante pour le joueur 1 lorsque n=8

Posons $\mathcal{A} = \{1, 2, 3, 4, 5, 6, 7, 8\}$ notre alphabet, l'ensemble des mots de départ est alors $\mathcal{D} = \{2, 4, 6, 8\}$. Voici l'arbre du jeu lorsque le joueur 1 joue 2 comme premier coup :



Le coloriage du graphe nous montre que 2 appartient à U. En commençant par jouer 2, J_1 a donc une stratégie gagnante.

VI.3 Récapitulatif des stratégies gagnantes

Voici un tableau récapitulant quel joueur possède une stratégie gagnante selon la valeur de n.

Gagnant	valeur de n
J_1 gagne	3-8-12-13-14-16-17-20-24-27-30-31-33-34-36-37-38-39-40-41-44-48-49-50
J_2 gagne	2-4-5-6-7-9-10-11-15-18-19-21-22-23-25-26-28-29-32-35-42-43-45-46-47

On crée ainsi deux suites : la suite des rangs pour lesquels J_1 a une stratégie gagnante et celle des rangs pour lesquels J_2 a une stratégie gagnante.

Une question se pose : Pourrions nous connaitre pour n quelconque si la stratégie gagnante est pour J_1 ou pour J_2 ? Une réponse positive à cette question semble compromise pour deux raisons. Premièrement il ne semble pas y avoir de schéma répétitif évident pour aucune des deux suites, et deuxièmement le calcul des termes suivants de chacune de ces suites croit de manière exponentielle.

VII Algorithme de recherche de stratégie gagnante

Maintenant que l'on est assuré de l'existence d'une stratègie gagnante pour l'un des deux joueurs, on peut déterminer l'existence d'un piège ou d'une stratègie gagnante avec les fonctions à récursivités croisées suivantes :

```
fonction existePiege(partie : mot):bouléen
    Si dernierCoup = 1 alors retouner vrai
    existe <- faux
    coup <-  1
    Tant Que (existe=faux & coup <= nbMax)
       Si coup est Valide alors
           ajoutePartie(coup)
           Si gagnant(partie) alors  existe <- vrai
           enlevePartie(coup)
           coup <- coup + 1
    Fin Tant Que
    retourner existe

Fonction gagnant(partie :mot):bouléen
    Si dernierCoup=1 alors retourner faux
    coup <- 1
    gagne <- vrai
    Tant Que (coup <= nbMax & gagne)
        Si coup est Valide alors
            ajoutePartie(coup)
            Si existePiege(partie)=faux alors gagne <- faux
            enlevePartie(coup)
            coup <- coup + 1
    Fin Tant Que
    retourner gagne
```

Cette recherche optimisée permet aux ordinateurs de jouer jusqu'à nbMax=50 où partie est un tableau de 50 entiers.

VIII Conclusion

Lorsque pour la première fois nous nous sommes posé la question : " Comment fait-on pour gagner au jeu de Juniper Green ?", nous ne pensions pas répondre à cette question en disant : " Applique simplement la stratégie gagnante !"

Pourtant le résultat est là, pour n quelconque il existe une stratégie gagnante pour l'un des deux joueurs.

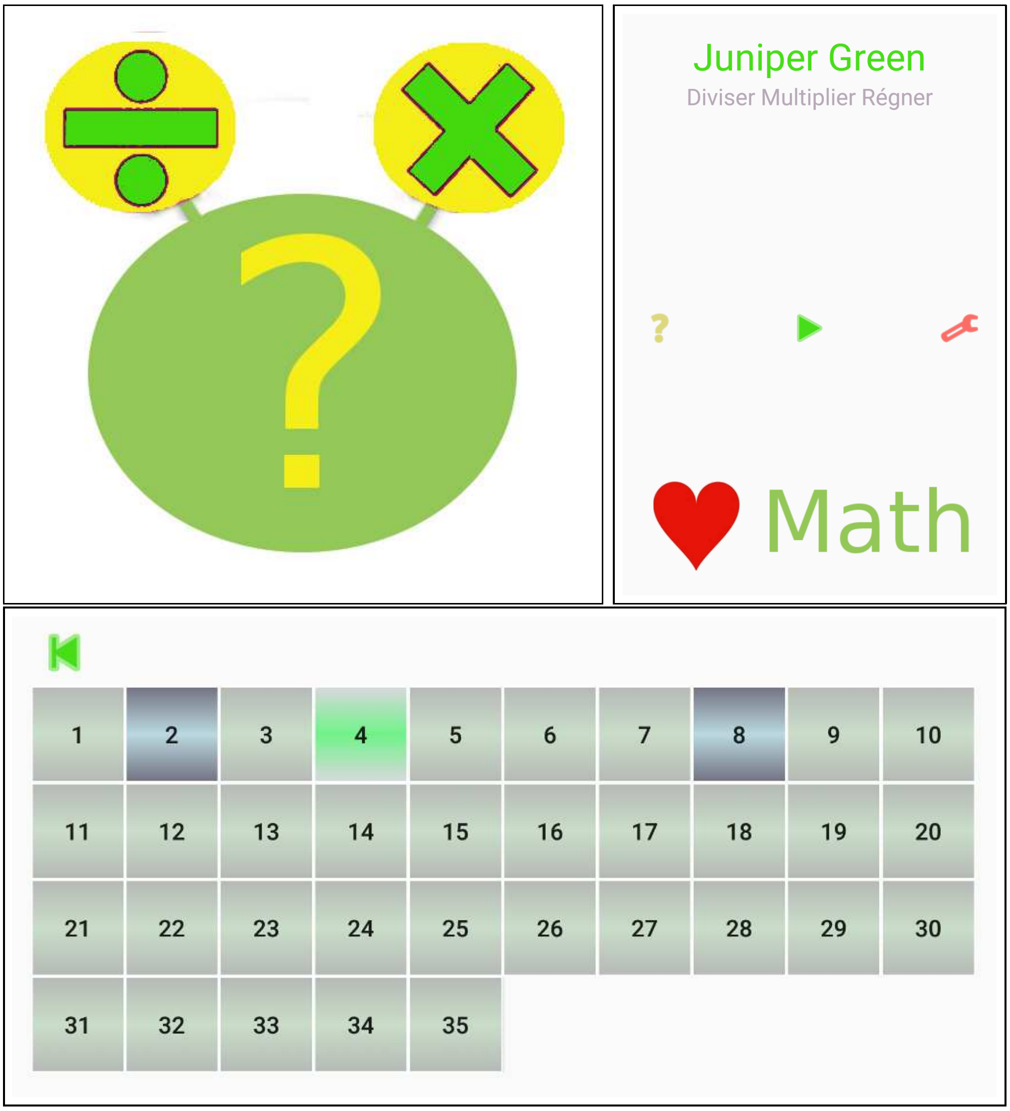
Nous tenons à signaler que dans cette aventure, les tâches ont été partagées et il est important de rendre à Caesar ce qui est à Caesar.

— Créateur du jeu sans les IA sous PYTHON : Olivier SICARD

— Créateur de l'application smartphone : Boris LAVAL
— Créateur des IA humaines : Olivier SICARD
— Créateur de l'IA inhumaine par approche probabiliste : Boris LAVAL
— Créateurs de l'IA avec stratégie gagnante basée sur la démonstration : Boris LAVAL et Olivier SICARD
— Créateur de l'algorithme de recherche optimisé : Boris LAVAL

IX Annexes

Application Android



https ://play.google.com/store/apps/details ?id=com.lavalpro.juniper