

Simulation et projet

Implémentation d'un
système multi agents
sur des bulles de savon



Table des matières

Introduction

1 Les Agents

1.1 Les bulles

1.2 Les chasseurs

2 choix de l'implémentation du SMA

2.1 Les bulles

2.2 Les chasseurs

3 Analyse du comportement du SMA

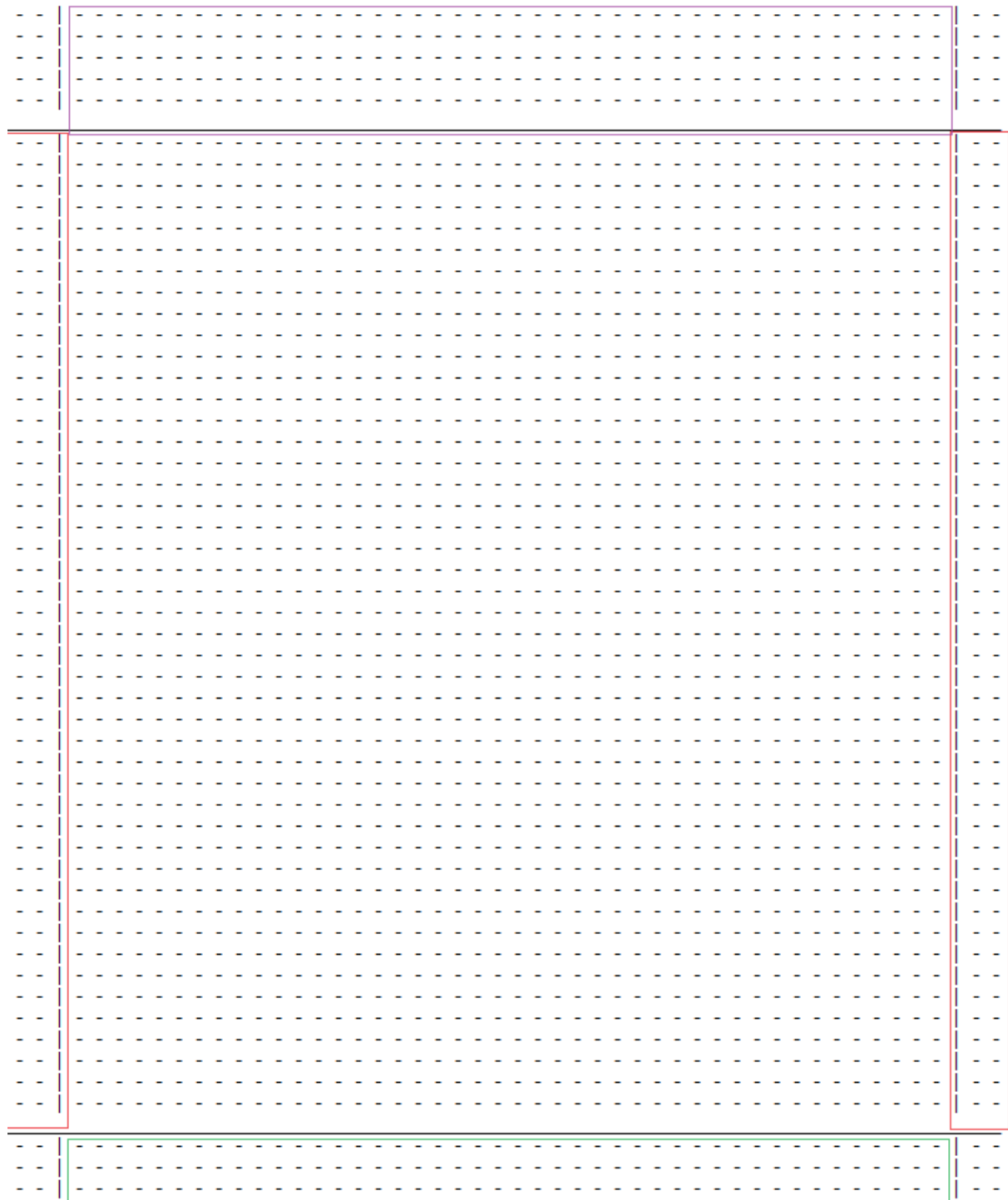
Conclusion

Introduction

Nous avons choisi de simuler une envolée de bulles de savons qui essaient d'échapper à des chasseurs qui ont pour but de les éclater. Les déplacements des bulles sont un peu hasardeux et les chasseurs sont restreints à leur zone de tir.

Ce dossier explique les fonctionnalités du SMA implémenté et présente quelques tests effectués sur son comportement.

Le monde :



Les bulles évoluent dans un univers représenté par un tableau ou un cylindre. Les bulles apparaissent sur le bas du tableau (cadre vert). Sur les côtés (cadres rouges) de ce monde, il y a des chasseurs qui essaient d'éclater les bulles. Les bulles s'envolent vers le haut et sont déclarées sauvées quand elles arrivent tout en haut (cadre violet).

La taille du monde est modifiable mais, pour les tests, nous avons utilisé un monde de taille 50 * 50 avec des rebords pour les chasseurs de taille 3 et un rebord pour l'apparition des bulles en bas de taille 5. Les Chasseurs sont déclarés vainqueurs s'ils ont tué plus de bulles qu'il n'y a eu de bulles sauvées, sinon les bulles gagnent.

Une partie se déroule de la façon suivante :

- on fait apparaître les chasseurs
- on fait X tours de jeu
- Si on met fin à la partie quand il n'y a plus de bulles, on effectue des tours sans apparitions de bulles tant qu'il reste des bulles

Un tour se déroule de la façon suivante :

- Toutes les bulles se déplacent
- On éclate les bulles suivant la méthode décidée (contact ou même case)
- Les chasseurs se déplacent puis tirent sur la bulle la plus proche
- Les bulles apparaissent dans leur zone

1 Les agents

1.1 Les bulles

Un nombre de bulles donné apparaît dans leur zone d'apparition au début de chaque tour, après les déplacements des bulles qui existent déjà.

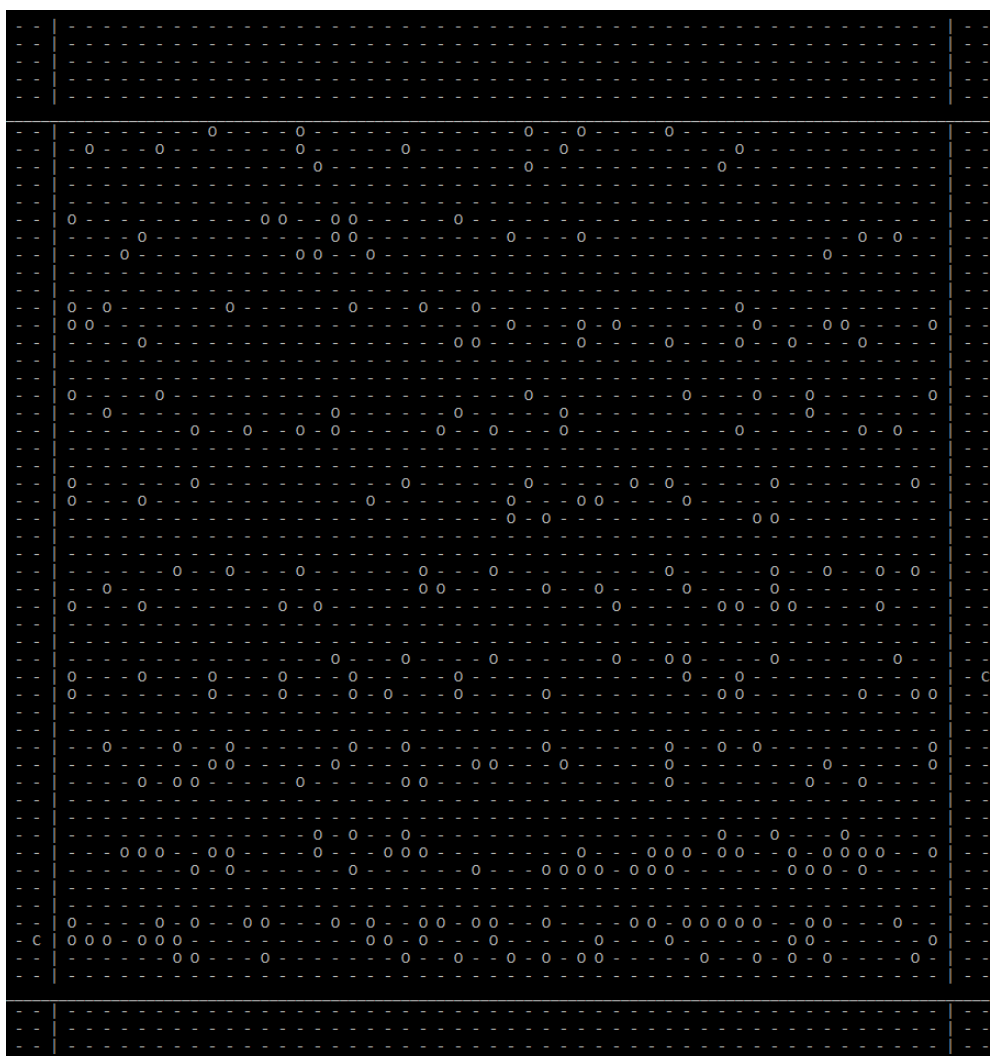
Les bulles se déplacent de manière hasardeuse mais toujours vers le haut.

Les bulles qui se déplacent sur la même case s'éclatent toutes les deux. Une bulles qui éclatent peut laisser tomber des éclaboussure et éclater toute bulles dans la même colonnes qu'elle si le mode éclaboussure est activé.

De même nous pouvons activer le mode contact où les bulles éclatent aussi quand elles sont sur des cases adjacentes.

1.2 Les chasseurs

Les chasseurs apparaissent n'importe où dans leur zone d'activité, au début de la partie. Les chasseurs se déplacent dans leur zone, vers le haut ou vers le bas, et n'importe où dans cette zone vers la gauche ou vers la droite. Après leur déplacement, chacun tire sur la bulle la plus proche de lui, cependant un chasseur peut manquer sa cible.



2 Implémentation du SMA

Notre choix pour l'implémentation du projet a été pour nous une évidence. En effet, l'utilisation du langage Java, orienté objet, paraissait bien plus adapté étant donné que les chasseurs et les bulles ont leurs propres caractéristiques. Que ce soit en termes de déplacement, d'apparition ou de comportement il est nécessaire de traiter les bulles et les chasseurs comme des objets distincts.

2.1 Les Bulles

Les bulles se déplacent d'une distance vers le haut et d'une distance variable sur le côté entre -1 et 1 par étape et il y a 5 étapes par tour.

```
50 // se deplace d'un vers le haut et dans une zone de droite a gauche
51 public void moveBulle()
52 {
53
54     this.y = this.y +1;
55     this.x = this.x + uniformSpawn(Main.DECAL_GAUCHE, Main.DECAL_DROITE);
56
57 }
```

A la fin de tous les déplacements, on arrondit la position des bulles à la case la plus proche. Les déplacer comme ceci permet d'avoir moins de variance dans les déplacements, tandis que si nous avions directement déplacé les bulles de 5 vers le haut et dans une fourchette entre -5 et 5 sur les côtés les bulles se seraient beaucoup trop déplacées et n'auraient pas eu un comportement très représentatif du comportement réel d'une bulle.

De plus, on peut ainsi changer facilement le nombre d'étape dans les paramètres. Pour supprimer les bulles nous avons choisi d'ajouter un boolean dans les caractéristiques des bulles de valeur false quand on veut supprimer toutes les bulles d'un seul coup et pouvoir également avoir un visuel sur les bulles qui vont disparaître.

2.2 Les Chasseurs

Les chasseurs tirent sur la bulle la plus proche d'eux à chaque tour, et après s'être déplacé, on utilise à ce moment la variable precision

```
56 if ( ( (distance - 1)* Main.PRECISION )/100 < (MersenneTwisterFast.getMT().nextDouble()) )
```

$((\text{distance} - 1) * \text{precision}) / 100$ est la probabilité du chasseur de manquer sa cible. donc plus la précision est basse plus le chasseur a de chances de réussir, même chose pour la distance, plus une bulle est proche plus le chasseur a de chances de la toucher.

3 Analyse du comportement du SMA

Tous les résultats des tests effectués sont obtenus après une simulation de 50 000 parties .

Simulations témoins par quantité de bulles qui apparaissent par tour.

win bulle	0,000%	0,546%	96,612%	100,000%
win chasseur	100,000%	99,454%	3,388%	0,000%
nb bulle	2	4	8	16
précision	2	2	2	2
cylindre	non	non	non	non
contact	non	non	non	non
éclaboussure	non	non	non	non
fin	non	non	non	non

On peut voir que la quantité de bulles qui apparaissent est très importante :

- pas assez et les bulles ne peuvent pas gagner
- trop et elles ne peuvent pas perdre.

Simulations sur l'importance de la précision du chasseur.

win bulle	0,000%	0,000%	38,764%	100,000%
win chasseur	100,000%	100,000%	61,236%	0,000%
nb bulle	2	4	8	16
précision	0,01	0,01	0,01	0,01
cylindre	non	non	non	non
contact	non	non	non	non
éclaboussure	non	non	non	non
fin	non	non	non	non

Une meilleure précision du chasseur augmente grandement sa réussite mais les chasseurs sont quand même impuissants contre une grande population de bulles.

Simulations sur l'efficacité du monde cylindrique par rapport au monde avec murs.

win bulle	0,000%	0,558%	96,332%	100,000%
win chasseur	100,000%	99,442%	3,668%	0,000%
nb bulle	2	4	8	16
précision	2	2	2	2
cylindre	oui	oui	oui	oui
contact	non	non	non	non
éclaboussure	non	non	non	non
fin	non	non	non	non

Le monde cylindrique n'as que très peu d'impact sur les résultats de la simulation, ce changement est négligeable sur une grande quantité de simulation.

Simulations avec collision entre le bulles.

win bulle	0,000%	0,062%	12,670%	67,762%	77,974%
win chasseur	100,000%	99,938%	87,330%	32,238%	22,026%
nb bulle	2	4	8	16	50
précision	2	2	2	2	2
cylindre	non	non	non	non	non
contact	oui	oui	oui	oui	oui
éclaboussure	non	non	non	non	non
fin	non	non	non	non	non

L'ajout des collisions amoindri grandement l'efficacité des bulles car les bulles ont besoins d'être nombreuses pour gagner mais plus elles sont nombreuses plus elles prennent le risque d'entrer en contact, et donc de s'éclater. On peut voir que, par rapport aux tests précédents, les bulles n'arrivent jamais à 100% de chance de gagner.

Simulations avec projections d'éclaboussures lorsqu'une bulle éclate.

win bulle	0,000%	0,348%	83,936%	99,994%
win chasseur	100,000%	99,652%	16,064%	0,006%
nb bulle	2	4	8	16
précision	2	2	2	2
cylindre	non	non	non	non
contact	non	non	non	non
éclaboussure	oui	oui	oui	oui
fin	non	non	non	non

Les éclaboussures freinent un peu les bulles mais ce changement est de loin inférieur au changement provoqué par les collisions.

Simulations avec fin de la partie dès qu'il n'y a plus de bulles.

win bulle	0,000%	43,896%	100,000%	100,000%
win chasseur	100,000%	56,104%	0,000%	0,000%
nb bulle	2	4	8	16
précision	2	2	2	2
cylindre	non	non	non	non
contact	non	non	non	non
éclaboussure	non	non	non	non
fin	oui	oui	oui	oui

Ce changement augmente considérablement la réussite des bulles, les chasseurs ont déjà du mal à partir de 4 bulles par tour.

Simulations avec collisions et fin dès qu'il n'y a plus de bulles.

win bulle	0,000%	6,482%	93,300%	99,978%
win chasseur	100,000%	93,518%	9,700%	0,022%
nb bulle	2	4	8	16
précision	2	2	2	2
cylindre	non	non	non	non
contact	oui	oui	oui	oui
éclaboussure	non	non	non	non
fin	oui	oui	oui	oui

Les collisions ne suffisent pas à compenser le changement de fin de partie mais presque.

Simulations avec collisions, éclaboussures et fin dès qu'il n'y a plus de bulles.

win bulle	0,000%	3,382%	77,240%	98,488%
win chasseur	100,000%	96,618%	22,760%	1,512%
nb bulle	2	4	8	16
précision	2	2	2	2
cylindre	non	non	non	non
contact	oui	oui	oui	oui
éclaboussure	oui	oui	oui	oui
fin	oui	oui	oui	oui

Avec les éclaboussures en plus des collisions, le changement de fin a été plus que compensé. Malgré cela, les chasseurs n'arrivent toujours pas à avoir de bonnes chances de victoire contre des bulles en grande quantité.

Simulations avec collisions, éclaboussures, meilleure précision des chasseurs, et fin dès qu'il n'y a plus de bulles.

win bulle	0,000%	0,000%	2,540%	50,814%	62,212%
win chasseur	100,000%	100,000%	97,460%	49,186%	37,788%
nb bulle	2	4	8	16	50
précision	0,01	0,01	0,01	0,01	0,01
cylindre	non	non	non	non	non
contact	oui	oui	oui	oui	oui
éclaboussure	oui	oui	oui	oui	oui
fin	oui	oui	oui	oui	oui

Avec ces paramètres, les chasseurs ont ½ de gagner à 16 bulles et même à 50 bulles ils ont 37% de chances de gagner ce qui est bien meilleur que les résultats précédents.

Simulations avec collisions, éclaboussures et une très bonne précision pour les chasseurs.

win bulle	0,000%	0,000%	0,012%	5,982%	10,472%
win chasseur	100,000%	100,000%	99,988%	94,018%	89,528%
nb bulle	2	4	8	16	50
précision	0,01	0,01	0,01	0,01	0,01
cylindre	non	non	non	non	non
contact	oui	oui	oui	oui	oui
éclaboussure	oui	oui	oui	oui	oui
fin	non	non	non	non	non

Si on donne tout les avantages aux chasseurs, ils arrivent à monter à 89% de chances de gagner contre 50 bulles par tour donc on peut largement compenser la force du nombre de bulles en utilisant les bons paramètres.

Conclusion

Ce SMA nous a permis de mieux saisir les intérêts et enjeux des simulations. En effet, étant donné une situation, il est possible d'anticiper des résultats et de pouvoir faire des tests virtuels bien trop longs, ou coûteux, ou dangereux, ou complexes à mettre en place réellement.

Les simulations permettent aussi d'ajuster des paramètres en se basant directement sur les conséquences de leurs changements, comme nous l'avons montré dans la dernière partie de ce dossier. On en déduit qu'il s'agit probablement du genre de méthodes utilisées dans le développement des jeux vidéos.