

Détection d'événements rares dans les simulations multi-agents

Yu-Lin Huang
ylin.huang@univ-artois.fr

Gildas Morvan
gildas.morvan@univ-artois.fr

Frédéric Pichon
frederic.pichon@univ-artois.fr

David Mercier
david.mercier@univ-artois.fr

Univ. Artois, EA 3926, Laboratoire de Génie Informatique et d'Automatique de l'Artois (LGI2A), Béthune, France

Résumé

De nombreuses simulations multi-agents sont des processus stochastiques, pouvant induire une forte variabilité. Cet article introduit une nouvelle politique d'exécution de simulations multi-agents, basée sur la sélection et le clonage de l'état des simulations, dont l'objectif est de biaiser la distribution originale des solutions pour générer des événements rares en exécutant un minimum de simulations. Cette politique est validée sur la base d'exemples de la littérature (marche aléatoire, modèle proie-prédateurs).

Mots-clés : simulation multi-agents, événements rares, politique d'exécution

Abstract

Many multi-agent based simulations are stochastic processes that can induce high variability. This article introduces a novel execution policy of multi-agent based simulations, based on the selection and cloning of the simulation state, whose purpose is to generate rare events while executing a minimal number of simulations. This policy is validated on the basis of examples from the literature (random walk, prey-predator model).

Keywords: multiagent simulation, rare events, execution policy

1 Introduction

Imaginons un biologiste étudiant un réseau trophique et disposant, à des fins de prédiction, d'un très bon modèle multi-agents stochastique de celui-ci. Notre biologiste souhaite répondre à la question suivante : « Quelle est la probabilité qu'une ou plusieurs espèces du réseau disparaissent d'ici un certain temps ? »

L'approche classique pour y répondre consiste à utiliser la méthode de Monte Carlo [5] : le modèle est simulé n fois pour estimer les probabilités d'apparition des différentes solutions. Pour

déterminer la valeur minimale de n permettant des résultats statistiquement significatifs, l'algorithme suivant est généralement appliqué [1] :

1) Exécuter n_0 simulations. 2) Calculer

$$n = \lceil \max_{o \in O} \left(\frac{Z_{1-(\alpha/2)} \cdot s_0(o)}{\epsilon \cdot \bar{o}_0} \right)^2 \rceil \quad (1)$$

où o est un observable, O l'ensemble des observables, ϵ l'erreur relative voulue de l'estimation, $1 - \alpha$ la probabilité que l'erreur relative réelle de l'estimation soit bien inférieure à ϵ , $Z_{1-(\alpha/2)}$ le $1 - \alpha$ quantile de la distribution normale, $s_0(o)$ l'écart-type de o et \bar{o}_0 sa moyenne. Les valeurs généralement utilisées pour n_0 , ϵ et α sont respectivement 150, 0.05 et 0.05. 3) Si $n > n_0$, exécuter les $n - n_0$ simulations restantes.

Notons que l'utilisation de la fonction \max est ici nécessaire pour pouvoir estimer correctement la valeur de l'observable ayant la plus grande variabilité.

Cette méthode est-elle applicable dans le cas général ? Les travaux menés sur le sujet répondent par la négative [4]. Si certaines solutions ont une faible probabilité d'apparition, mais que l'on souhaite néanmoins les détecter, une importante puissance de calcul sera nécessaire. Afin de pallier ce problème, nous proposons dans cet article une nouvelle politique d'exécution de simulations multi-agents. L'objectif est de biaiser la distribution originale des solutions afin de favoriser certaines, puis de la reconstruire *a posteriori* en exécutant un minimum de simulations.

Dans cet article, nous nous focalisons sur la première étape où il s'agit de biaiser la distribution originale. Certains aspects de ce travail, décrits succinctement pour des raisons d'espace (méthodes de sélection, clonage des simulations), seront explicités dans de futures publications.

2 Présentation de la méthode

Les quatre idées principales qui sous-tendent notre approche sont : **1)** une simulation peut être, à chaque pas de temps, dans plusieurs états simultanément, **2)** un état peut être cloné (répliqué) et réinjecté dans une nouvelle simulation, **3)** une simulation peut-être échantillonnée dans le temps et ainsi *découpée* en plusieurs morceaux, **4)** parmi les différents états, il est possible de sélectionner à intervalle régulier les plus intéressants pour répondre à la question posée par le modélisateur. Ce processus de sélection peut viser par exemple à obtenir les résultats les plus divers possibles ou correspondant à certains critères.

La politique d'exécution est décrite dans la figure 1. k états initiaux sont répliqués n fois puis simulés sur t_s pas de temps. Les k plus intéressants sont sélectionnés puis à nouveau répliqués. Le processus se poursuit jusqu'au pas de temps final.

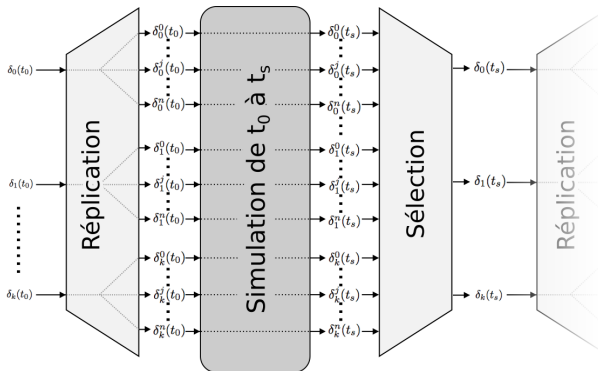


FIGURE 1 – Algorithme proposé

La principale contrainte opérationnelle imposée par cette politique est de pouvoir cloner les simulations. Dans le cas général, ce peut-être un processus coûteux, notamment si le nombre d'agents est important. L'architecture du simulateur utilisé pour les expérimentations (Similar2Logo¹) repose sur l'idée d'une séparation déclaratif/procédural permettant de simplifier le clonage [3]. L'état d'une simulation est ainsi « isolé » et peut-être rapidement injecté dans une nouvelle simulation, induisant un surcoût négligeable.

1. <https://github.com/gildasmorvan/similar2logo>

3 Travaux connexes

La politique d'exécution présentée dans cet article peut-être rattachée à différentes approches existantes.

Tout d'abord, les diverses techniques de *splitting* utilisées pour calculer la probabilité qu'un événement rare survienne dans une simulation d'un modèle de Markov [2, 8]. Ces techniques assument des hypothèses fortes sur les propriétés du processus simulé et sur la nature des événements rares : il est ainsi nécessaire de connaître la chaîne d'événements (moins rares) conduisant à celui-ci [4]. Ces contraintes semblent difficilement satisfaites dans le cas général pour un modèle multi-agents. D'un point de vue algorithmique, la différence majeure entre les techniques de *splitting* et l'approche que nous proposons, est la nature de l'échantillonnage : dans l'espace des solutions dans le premier cas, temporel dans le second.

Dans le domaine de la simulation multi-agents, la politique d'exécution basée sur le concept de *polyagent* [7, 6]. L'objectif de cette technique est de caractériser rapidement la trajectoire moyenne d'une simulation. A intervalle régulier, les agents génèrent des *ghosts* reproduisant leur comportement et explorant l'espace des trajectoires possibles en émettant des phéromones. L'agent remonte ensuite le gradient du champ ainsi calculé jusqu'à son maximum, qui définira l'état de l'agent. La principale différence entre le concept de *polyagent* et l'approche que nous proposons est le niveau de sélection : dans le premier cas, des trajectoires individuelles sont sélectionnées, dans le second, des trajectoires collectives.

4 Exemples d'application

Nous allons dans cette section montrer l'intérêt de la politique d'exécution proposée en l'appliquant à des modèles académiques simples (marche aléatoire et proies-prédateurs).

4.1 Marche aléatoire

Le cas étudié ici est le plus simple possible : un agent se déplaçant selon une marche aléatoire suivant une loi normale de moyenne $\mu = 0$ et d'écart type $\sigma = 0.1$. L'événement rare considéré ici est le chemin menant à des valeurs les plus éloignées de l'origine possibles.

Par exemple, si nous cherchons les valeurs les plus petites possibles, la politique d'exécution consiste à sélectionner la simulation dont la position est minimale parmi l'ensemble de simulations puis la dupliquer n fois.

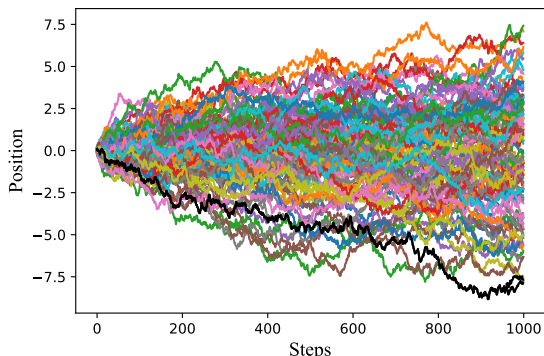


FIGURE 2 – Réalisations de la marche aléatoire avec $\sigma = 0.1$. Les courbes colorées représentent les réalisations avec l'approche classique, la courbe noire avec l'approche proposée.

La figure 2 montre des résultats obtenus par l'approche classique (courbes colorées, 100 simulations) et une réalisation avec l'approche proposée (courbe noire, $n = 2$, simulation échantillonnée tous les 50 pas de temps). Nous observons ici qu'avec seulement $n = 2$, le résultat obtenu par notre approche permet de biaiser la distribution originale des solutions pour obtenir la valeur quantile $q_{0.01}$ estimée par 100 simulations lancées classiquement.

4.2 Modèle proies-prédateurs

Nous considérons ici un modèle proies-prédateurs minimal, constitué d'agents « prédateurs » chassant des agents « proies », eux-même se nourrissant de la végétation disponible dans l'environnement.

Les régimes stationnaires (solutions) pouvant être obtenus avec ce modèle sont les suivants : R_1 : survie des proies et prédateurs, l'évolution des populations étant quasi périodique, R_2 : extinction des prédateurs et survie des proies, R_3 : extinction des proies et des prédateurs.

La probabilité d'obtention de ces régimes varient en fonction des paramètres du modèle. Dans la suite de cet article, nous considérons un paramétrage où les probabilités d'obtention des régimes R_2 et R_3 sont largement inférieures au régime R_1 ($P(R_1) \simeq 0.974$, $P(R_2) \simeq 0.025$ et $P(R_3) \simeq 0.001$). En lançant n simulations de la

manière classique, la probabilité de ne pas obtenir un régime R_i est ainsi égale à $(1 - P(R_i))^n$. Par exemple, si $n = 50$ nous aurons 95% de chance de n'avoir que des simulations en régime R_1 et R_2 .

Sélection multi-critères. Pour appliquer la méthode proposée précédemment, il faut d'abord définir la méthode de sélection la plus appropriée. Si nous cherchons à obtenir les 3 régimes, il semble naturel de sélectionner les états selon les trois critères correspondants : 1) La norme $\|\cdot\|_2$ maximale du couple nombre de proies, nombre de prédateurs. 2) Le nombre de prédateur minimum qui en même temps possède le nombre de proies maximum. 3) Le nombre de proies minimum qui en même temps possède le nombre de prédateur maximum.

Au moment du découpage, les simulations sont sélectionnées selon ces critères et dupliquées $\lfloor \frac{n}{3} \pm 1$ fois.

Notons que cette méthode de sélection suppose une connaissance *a priori* des régimes.

Enveloppe convexe. Une autre façon d'envisager la sélection, sans introduire de connaissance *a priori*, est d'identifier les simulations les plus différentes. Pour ce faire, nous calculons l'enveloppe convexe qui nous donne un sous-ensemble convexe dans la géométrie des observables des simulations. Nous sélectionnons selon le plan $n_{proies}/\Delta n_{proies}$, puis les clonons $\lfloor \frac{n}{k} \pm 1$, où k est le nombre de simulations sélectionnées.

Résultats. Pour évaluer les deux méthodes proposées, nous allons les comparer avec la méthode classique (référence). La figure 3 montre la probabilité d'obtenir au moins une simulation de l'événement rare R_3 en lançant n simulations avec différentes méthodes de sélection et différentes valeurs de n et une fréquence d'échantillonnage de 0.2. Ces résultats sont comparés avec la méthode de référence (Monte Carlo).

5 Discussion

Les résultats obtenus par la politique d'exécution proposée semble la valider : elle permet ainsi, dans le cas des deux exemples présentés, de biaiser la distribution originale des solutions et de générer ainsi des événements rares en exécutant un nombre réduit de simulations, même sans disposer de connaissances *a priori*

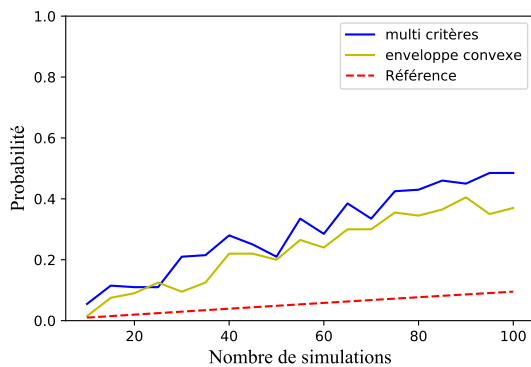


FIGURE 3 – Probabilité de détecter l’extinction des proies et des prédateurs (régime R_3) en exécutant n simulations

sur la nature des solutions possibles (sélection par enveloppe convexe). Très logiquement, cette méthode de sélection « à l’aveugle » donne de moins bons résultats que la méthode de sélection multi-critères intégrant ces informations. Notons également que la méthode de sélection par enveloppe convexe peut être étendue, en ne considérant plus seulement les observables, mais également leurs dérivées successives.

6 Conclusion

Dans cet article, nous avons proposé une nouvelle politique d’exécution de simulations multi-agents, puis avons établi les liens avec des travaux connexes. Nous avons ensuite proposé des éléments de validation sur la base de deux exemples. Nous avons ainsi montré que des méthodes simples de sélection permettent de biaiser efficacement la distribution originale des solutions pour générer des événements rares.

Dans la suite de ce travail, nous chercherons à définir des méthodes de sélection plus intelligentes en nous inspirant de techniques d’apprentissage non supervisé de séries temporelles afin de sélectionner de façon plus générale des classes de résultats. Les principaux verrous scientifiques étant ici le choix des méthodes de transformation des données (*e.g.*, en ondelettes) et de classification (*e.g.*, K-means).

Par ailleurs, nous n’avons pas abordé ici le problème du paramétrage de notre algorithme. Les expérimentations présentées dans cet article ont été conduites avec des valeurs de n , k et une fréquence d’échantillonnage fixées arbitrairement. L’un des objectifs de cette recherche sera de déterminer les valeurs optimales pour ces dif-

férents paramètres selon les caractéristiques du modèle simulé.

Enfin, et c’est là la principale perspective de ce travail, nous visons à reconstruire la distribution originale des solutions, éventuellement avec une certaine imprécision, pour estimer la probabilité d’apparition des événements rares générés.

Remerciements

Les travaux présentés dans cet article ont été réalisés dans le cadre du projet ELSAT2020, cofinancé par l’Union Européenne avec le Fonds européen de développement régional, par l’État et la Région Hauts-de-France.

La figure 1 a été réalisée par Yoann Kubera.

Références

- [1] J. Banks, J.S. Carson II, B.L. Nelson, and D.M. Nicol. *Discrete-event system simulation*. Pearson, 5th edition, 2010.
- [2] H. Kahn and T.E. Harris. Estimation of particle transmission by random sampling. *National Bureau of Standards applied mathematics series*, 12 :27–30, 1951.
- [3] Y. Kubera, P. Mathieu, and S. Picault. IODA : an interaction-oriented approach for multi-agent based simulations. *Autonomous Agents and Multi-Agent Systems*, 23(3) :303–343, 2011.
- [4] P. L’Ecuyer, F. Le Gland, P. Lezard, and B. Tuffin. *Rare Event Simulation using Monte Carlo Methods*, chapter Splitting Techniques. Wiley, 2009.
- [5] N. Metropolis and S. Ulam. The monte carlo method. *Journal of the American statistical association*, 44(247) :335–341, 1949.
- [6] H.V.D. Parunak. Pheromones, probabilities and multiple futures. In *Multi-Agent-Based Simulation XI*, volume 6532 of *LNCS*, pages 44–60. Springer, 2011.
- [7] H.V.D. Parunak and S. Brueckner. Concurrent modeling of alternative worlds with polyagents. In *Multi-Agent-Based Simulation VII*, *LNCS*, pages 128–141. Springer, 2007.
- [8] M. Villén-Altamirano and J. Villén-Altamirano. Restart : a straightforward method for fast simulation of rare events. In *Proc. of the Winter Simulation Conf. (WSC)*, pages 282–289, 1994.