

# **Modélisation et Simulation MultiAgents**

## Rapport de Projet

Maxime Desbois et Simon Lassourreille

Décembre 2018

# I – Hypothèses d’implémentation

---

## H1 – VALEUR ACCORDÉE À UNE UTILITÉ

Dans le modèle, après toute interaction  $i$  entre  $a$  et  $b$ , l’agent  $a$  crée un rating  $r$  pour chacun des critères  $c$  évaluable dans cette interaction. Ce rating prend la forme d’un tuple  $r = (a, b, c, i, v)$  où  $v \in [-1, 1]$  est la valeur que  $a$  donne à  $b$  pour le critère  $c$  lors de l’interaction  $i$ .

Cependant, il n’est nulle part précisé comment l’agent  $a$  détermine cette valeur. Nous savons que dans l’implémentation de l’article,  $a$  reçoit une utilité  $\mathbf{ug} \in [-10, 10]$ . Nous ferons donc les hypothèses suivantes :

- On peut exprimer  $v$  comme une fonction de l’ $\mathbf{ug}$  obtenue par  $a$  lors de cette interaction, et cette fonction est propre à  $a$  :

$$v = f_a(\mathbf{ug})$$

- Au vu du manque d’informations, on fera l’hypothèse que cette fonction normalise  $\mathbf{ug}$  dans l’intervalle de  $v$ , et qu’elle est la même pour chaque agent :

$$f_a(x) = \frac{x}{10}, \forall a$$

Cette hypothèse est triviale, mais pourrait représenter une opportunité d’améliorer le modèle : tous les agents n’ont pas une appréciation linéaire de l’utilité. De même, cette appréciation n’est pas toujours objective. Dans des situations réalistes, certaines personnes notent trop sévèrement, alors que d’autres peuvent être trop indulgentes. Il serait donc intéressant de doter les agents d’utilités non-linéaire dans un premier temps, et peut être par la suite de différencier certains profils d’agents pour intégrer ces disparités dans les notes des différents critères.

Cette amélioration (assez simple à mettre en place) correspond à des comportements réels observés (la subjectivité) et peut venir se placer en complément / extension des comportements de mensonges, qui sont difficiles à distinguer d’un point de vue extérieur.

## H2 – DÉGRADATION DU SERVICE LIÉE À LA DISTANCE

L’article évoque une dégradation linéaire du service lorsque la distance entre les deux agents en interaction,  $a$  et  $b$ , dépasse la portée opérationnelle  $r_o(b)$  de l’agent fournissant le service. Cependant, les détails de cette dégradation linéaire ne sont pas donnés par l’article. Nous allons décrire ci-dessous notre réflexion pour traiter cette hypothèse.

L’utilité perçue est donc une fonction de  $x = d(a, b)$  la distance entre  $a$  et  $b$ . Dans un premier temps, on sait que cette fonction sera constante sur  $0 \leq x \leq r_o$ .

$$f_{UG}(x) = \begin{cases} UG, & \text{si } x \leq r_o(b) \\ UG + \alpha * (x - r_o), & \text{sinon} \end{cases}$$

Avec  $\alpha < 0$  pour obtenir une dégradation.

Cependant, il reste quelques problèmes : avec une expression aussi générale, on peut obtenir  $u < UG_{min} = -10$ , ce qui n’est pas censé être possible. Ici, deux options s’offrent à nous :

- Etablir un coefficient constant  $\alpha$  de dégradation, peu importe l’utilité émise par l’agent. Ce qui veut dire que la portée maximale  $r_{max}$  où  $u \leq -10$  varie d’une interaction à l’autre.
- Etablir une portée maximale fixe,  $r_{max}$ , et s’arranger pour que chaque agent situé entre  $r_o$  et  $r_{max}$  reçoive une utilité  $u \leq -10$ , pour toute interaction.

La première option est intéressante, mais pose le problème du choix du coefficient. Il est arbitraire mais assez peu de choix logiques semblent s’imposer. La seconde option permet de simplement fixer  $r_{max}$ , potentiellement en fonction de  $r_o$ , et simplifie alors le paramétrage. Nous avons choisi la deuxième, mais nous détaillerons en fin de section l’impact final attendu de cette partie du modèle.

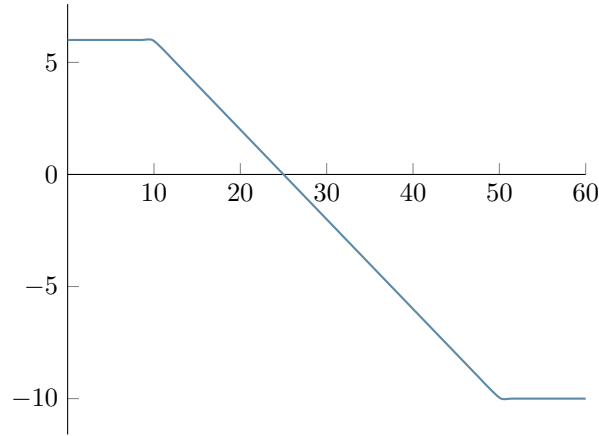
On ajoute donc la condition  $f_{UG}(x) = UG_{min}, \forall x \geq r_{max}$ . Ayant deux points de cette fonction linéaire, on peut alors déterminer que  $\alpha = \frac{UG_{min} - UG}{r_{max} - r_o}$ . Ce qui nous donne comme expression finale :

$$f_{UG}(x) = \begin{cases} UG, & \text{si } x \leq r_o(b) \\ UG + \alpha * (x - r_o), & \text{si } r_o \leq x \leq r_{max} \\ UG_{min}, & \text{si } x \geq r_{max} \end{cases}$$

Ou alors, en une seule expression,

$$f_{UG}(x) = \max(UG_{min}, \min(UG, UG + \frac{UG_{min} - UG}{r_{max} - r_o} * (x - r_o)))$$

On peut vérifier avec un graphe que ce comportement correspond à ce qui est attendu, avec :  $UG = 6$ ,  $UG_{min} = -10$ ,  $r_o = 10$ ,  $r_{max} = 50$



Cette méthode revient donc à dire que si  $a$  est à mi-distance entre  $r_o$  et  $r_{max}$ ,  $a$  recevra toujours  $\frac{u-10}{2}$ , au lieu de recevoir  $u - K$  avec  $K$  constant (à distance fixe).

Le choix de  $r_{max}$  est arbitraire, nous avons donc introduit un coefficient  $y0$  qui nous donne  $r_{max} = r_o * (1 + y)$ .

Nous faisons l'hypothèse que dans la figure 5 de l'article, une dégradation linéaire de la distance est utilisée. En effet, dans le cas où chaque provider est à 100% de ses capacités, l'utilité moyenne aléatoire estimée est de 1.5 pour les paramètres de la figure. Cependant, ils affichent une utilité aléatoire à -1, ce qui n'est pas réaliste si ils n'utilisent pas ce mécanisme.

$\mathcal{H}3$  –

## II – Pseudocode

```

INITIALISATION : CONSUMER
pour chaque consumer a faire
    initialiser une liste vide de taille H
    initialiser r_o
    choisir un niveau d'activité alpha ~ U([0.25, 1.00])
    choisir une position au hasard
fin pour

```

#### INITIALISATION : PRODUCER

```
pour chaque provider b faire
  déterminer le type de b parmi {good, ok, bad, intermitent}
  initialiser r_o
  si b n'est pas intermitent alors
    initialiser p
    choisir au hasard p dans l'intervale correspondant
  fin si
fin pour
```

#### DÉROULEMENT D'UN ROUND

```
pour chaque consumer a faire
  déterminer si a nécessite un service
  si vrai alors
    incrémenter i le compteur d'interactions
    b <-- choix d'un provider par a
    ug <-- b génère au hasard une utilité entre -10 et 10
    v <-- déterminer la valeur de l'interaction
    créer un rating r = (a, b, c, i, v)
  fin si
fin pour

# simulation d'instantanéité
pour chaque consumer a faire
  si r existe alors
    enregistrer le rating r dans la liste de a
    envoyer le rating r à b
    b enregistre r si v est meilleur que son moins bon référencement
  fin si
fin pour
```

#### CHOIX D'UN PROVIDER

```
P <-- récupérer les providers disponibles
pour chaque b dans P faire
  v <-- déterminer la valeur de confiance de b
  si v est défini alors
    insérer b dans la liste NoTrust sinon
    insérer b dans la liste Trust
  fin si
fin pour
trier la liste Trust par valeur de confiance
choisir stratégie (exploration ou exploitation)
si exploration alors
  p <-- choisir un provider au hasard dans NoTrust
sinon
  p <-- provider avec la plus grande valeur de confiance
fin si
retourner p
```

DÉTERMINER LA VALEUR DE CONFIANCE DE LA COMPOSANTE K  
Entrée: RK ensemble de ratings pour le calcul de la composante K

```
Σ <-- 0
Σ_vi <-- 0
pour chaque ri dans RK faire
  <-- calculer _K(ri)
  Σ <-- Σ +
  Σ_vi <-- Σ + * vi
fin pour
T_K <-- Σ_vi / Σ
_RK <-- 1 - exp(-K * Σ)
Σ_vi_T <-- 0
pour chaque ri dans RK faire
  Σ_vi_T <-- Σ_vi_T + _K(ri) * abs(vi - T_K)
fin pour
_DK = 1 - 0.5 * (Σ_vi_T / Σ)
_K = _RK * _DK

retourner (_K, T_K)
```

DÉTERMINER LA VALEUR DE CONFIANCE GÉNÉRALE  
# Déterminer RI l'ensemble des ratings d'Intéraction

### III – Etude préliminaire

---

**3.1** Résumez en quelque ligne la problématique des auteurs, et les choix de modélisation qu'ils proposent.

Les auteurs proposent un modèle de confiance permettant à un agent de juger de la qualité des services d'autres agents. Plus précisément, ce modèle doit être décentralisé, robuste aux mensonges.

**3.2** Pourquoi ? Test.

### IV – Etudes progressive du modèle FIRE

---

#### Conception de l'interface NetLogo

Est-ce qu'on en parle ? Si oui, c'est ici !

### V – Modèle avec composante IT seule (FIRE-IT)

---

**5.1** Écrivez en pseudo-code l'intégralité des algorithmes que vous allez implémenter pour simuler le modèle FIRE-IT. Expliquez et Justifiez vos choix (notamment pour l'implémentation de la topologie sphérique en NetLogo).

**5.2** Décrivez le protocole expérimental (et votre procédure d'initialisation des agents en particulier), et donnez dans un tableau la valeur de tous les paramètres que vous fixez pour reproduire la Figure 9.

## Initialisation des Agents

Comme discuté dans la section

**5.3** Affichez votre version de la Figure 9 et commentez les résultats.

## VI – Modèle avec composante IT et WR (FIRE-IT-WR)

---

**6.1** Écrivez en pseudo-code l'intégralité des algorithmes que vous allez implémenter pour simuler le modèle FIRE-IT-WR. Expliquez et Justifiez vos choix. Implémentez en Netlogo.

**6.2** Indiquez le protocole pour simuler la Figure, et les éventuels changements dans les paramètres (modifications des anciennes valeurs, éventuels ajouts de nouveaux paramètres)

**6.3** Affichez votre version de la figure 9 complète et expliquez les résultats.

## VII – Modèle avec composante IT, WR et CR (FIRE-IT-WR-CR)

---

**7.1** Écrivez en pseudo-code l'intégralité des algorithmes que vous allez implémenter l'ajout de la composante CR. Expliquez et Justifiez vos choix. Implémentez en Netlogo.

**7.2** Indiquez le protocole pour simuler ces Figures, et les éventuels changements dans les paramètres (modifications des anciennes valeurs, éventuels ajouts de nouveaux paramètres).

**7.3** Affichez vos version des Figures 10 et 11 et expliquez les résultats, notamment en comparaison de la Figure 9.

## VIII – Analyses de sensibilité du modèle FIRE-IT-WR-CR.

---

**8.1** Affichez un graphique 3D (avec un tableur ou grapheur) qui prend comme axes  $x$  et  $y$  les poids  $W_c$  et  $W_w$  et en axe  $z$  la performance moyenne des UG. Expliquez comment vous avez obtenu ce graphique et commentez les résultats.

**8.2** Affichez maintenant (avec les paramètres initiaux de [1]) un histogramme montrant la distribution des valeurs des UGs des clients. Commentez.

**8.3** Faites varier individuellement les paramètres  $NPG$ ,  $NPO$  et  $NB=NPI+NPB$  (mais en respectant toujours  $NPG+NPO+NB=100$ ). Dans des graphes séparés, afficher la moyenne et l'écart type des UG pour chacune 10 ces valeurs. Commentez. Quel est effet de ces paramètres sur la distribution des UG des clients ?

## IX – Extension –

---

## X – Bonus

---

**10.1** Expliquez en quelques lignes comment le modèle FIRE pourrait être adapté et étendu pour simuler une société telle qu'elle est décrite dans l'épisode Nosedive de la série Black Mirror.