

# RAPPORT FINAL

Ateliers de Recherche Encadrés – DYNAMIC

**Professeurs encadrants :**

Nicolas MAUDET

Jean Daniel KANT

**Equipe de développement :**

OGOLO Jenny

SADOUKI Aya

ZENG Ruxue

# « La diffusion d'une fake news dans un environnement donné et les conséquences de cette diffusion »

## Table des matières

Résumé : .....	1
Introduction : .....	2
Présentation de la thématique : .....	2
Développement : .....	3
- Code : .....	3
- Simulation : .....	5
- Analyse : .....	5
Conclusion : .....	6
Résumé en anglais : .....	6
Annexes : .....	7
- Graphiques : .....	7
- Spécification du modèle : .....	8
- Fonctions : .....	9

## Résumé :

Notre projet consiste en la modélisation et l'étude de la dynamique de propagation d'une fake news dans un environnement de **personnes** et d'**organismes médiatiques**. Nous avons ainsi voulu mettre en lumière l'influence de plusieurs paramètres sur l'ampleur et la vitesse de propagation d'une telle information.

Pour ce faire, les agents sont placés sur une grille à deux dimensions. Une fake news est instillée dans l'environnement et les agents peuvent interagir par discussion physique ou à distance s'ils sont reliés par un réseau social.

Les résultats obtenus sont sans appel. Nous verrons par exemple qu'une fake news se propage plus vite quand on augmente le nombre de réseaux sociaux moyen par agent.

*(Rédaction : Aya - relecture : Jenny)*

## Introduction :

Ce document est le rapport final du projet réalisé par Jenny OGOLO, Aya SADOUKI et Ruxue ZENG, configuré pour l'ARE DYNAMIC de l'Université Sorbonne Paris VI. Il a pour but d'expliquer notre travail réalisé pour le sujet « La diffusion d'une fake news ».

Aujourd'hui, nous sommes confrontés à une globalisation des fausses informations. Face à ce phénomène, le président Macron a d'ailleurs annoncé un projet de loi punissant la diffusion de fausses informations. Notre génération étant particulièrement active sur les réseaux sociaux, nous avons choisi de mener cette étude pour les enjeux qu'elles présentent. En effet, manipuler des personnes par le biais de l'information profère un intérêt économique et/ou politique non négligeable. Nous nous intéresserons donc ici aux fausses informations, appelées aussi « fake news ». Elles n'ont de nouveau que leur nom, puisqu'elles ont presque toujours existé. Cependant, l'avènement des réseaux sociaux et leur expansion planétaire ont fait exploser ce phénomène.

A partir de ce constat, nous allons donc essayer de réaliser un modèle, remplissant les conditions imposées par le cahier des charges établi précédemment. Pour ce faire, nous avons utilisé des logiciels que nous avons appris à manipuler durant cette année : Jupyter Notebook et MrPython, avec le langage Python. L'étude aurait pu être menée en utilisant d'autres logiciels tels que Spyder par exemple. Néanmoins, nous avons préféré des logiciels déjà manipulés auparavant, nous avons donc codé en Python principalement via Jupyter Notebook. Seule l'animation a été codée via MrPython.

En fonction des différents paramètres, comment la fake news va-t-elle se propager ? Qu'en est-il des personnes qui relaient délibérément une fausse information ? Comment pouvons-nous vérifier si notre modèle est concevable ou pas ?

Nous allons donc présenter la thématique de notre projet en explicitant les notions fondamentales sur lesquelles repose notre projet. Ensuite, nous allons développer le déroulement de notre système en expliquant petit-à-petit le code que nous avons écrit pour aboutir à des expériences que nous analyserons.

*(Rédaction : Ruxue - relecture : Aya)*

## Présentation de la thématique :

Les fake news (parfois appelées "fausses informations" ou "informations fallacieuses") contribuent à un phénomène de désinformation de plus en plus massif. Outre celles qui sont créées avec pour seul but de "faire le buzz", elles suscitent des enjeux politiques décisifs. Certains médias se sont d'ailleurs spécialisés dans la production de telles informations. Le tout est corroboré par une tendance globale à être connecté partout, tout le temps via les réseaux sociaux. Il devient alors évident que les médias et les réseaux sociaux seront au cœur de ce projet.

Pour assurer la propagation d'une fake news, il est essentiel que les agents impliqués y croient. C'est donc l'évolution de cette conviction que nous allons étudier. Cependant, on peut établir 2 types de convictions : la réponse qu'on obtient si l'on demande à quelqu'un s'il croit ou non à telle information (c'est ce que nous nommerons l'**opinion** qui est binaire) et l'indicateur qui permet de

quantifier si quelqu'un y croit plus ou moins intensément (nous l'appellerons la **conviction** et elle sera comprise entre 0 et 1). Ces deux notions, bien que dissociées, sont liées ; une opinion à 1 équivaut à une conviction supérieure à 0,5.

Ensuite, pour qu'un agent croie à une information, il faut que son transmetteur soit convaincant. Pour ce faire, on attribue à chaque agent une **force de persuasion**. Celle-ci est une qualité intrinsèque à chaque individu et quantifie la tendance qu'a un agent à faire tendre la conviction d'autrui vers la sienne. Dans une certaine mesure, elle pourra également servir à qualifier la "naïveté" d'un agent (nous le verrons plus tard avec la fonction "credib\_news"). En effet, une force de persuasion faible fera que son conviction pèsera moins et qu'il sera beaucoup plus enclin à tendre vers la conviction des autres.

Nous avons précédemment mentionné l'existence de médias spécialisés dans la production de fake news. Nous les avons intégrés dans notre modèle : ce sont les médias **malveillants**. Cet adjectif (quoiqu'un peu subjectif ...) est dû au fait qu'un organisme médiatique malveillant va, contrairement à un média classique, relayer une fake news en ayant totalement conscience qu'elle est fausse. Aussi, ce sont eux (dans le cadre de cette étude) qui sont à l'origine de la fake news. Lorsque nous parlerons de "malveillance", ce sera donc pour évoquer ce type de médias.

(*Rédaction* : Aya – *relecture* : Ruxue)

## Développement :

Passons maintenant à la partie technique du rapport : l'explication des grandes lignes du code.

### - Code :

L'écriture du code (outre la simulation) peut se décomposer en plusieurs parties : l'initialisation du système, la partie discussion physique et l'interaction à distance.

Nous avons commencé par modéliser la population par un **dictionnaire** qui à chaque agent associe un quadruplet (*int, int, float, float*) représentant respectivement l'opinion, le nombre d'expositions, la force de persuasion et la conviction d'un agent. Celui-ci suffit à décrire intégralement un agent et ses caractéristiques (voir description d'un agent en annexe). Nous avons ensuite disposé tous les agents (leur nom) sur une **matrice** de taille  $N_{1dim} * N_{1dim} = N$  individus. Nous avons également pris soin de représenter sur deux autres matrices de même taille leur opinion ainsi que leur conviction. *codé par Aya*

Concernant les réseaux sociaux, nous avons mis la connectivité en paramètre. Il nous fallait donc trouver un moyen d'établir le nombre de réseaux sociaux de chaque agent en connaissant le nombre de réseaux moyen par agent. Après réflexion, nous avons décidé d'établir une liste de N individus en choisissant les nombres deux par deux : le premier est choisi au hasard (on l'appellera "*nb*") et le suivant vaut  $2 * co - nb$  (si N est impair, on rajoute *co* à la fin). *codé par Ruxue* Chaque agent a ainsi une liste de réseaux parmi les 10 réseaux les plus utilisés en 2017. On a ensuite établi un **réseau virtuel** : pour chaque agent on associe une liste d'abonnés parmi les utilisateurs des réseaux sur lesquels il est inscrit. *codé par Aya*

Pendant une partie du code, nous nous sommes focalisées sur la propagation d'une fake news par le seul biais d'une interaction entre agents physiquement proches (dans la matrice). A partir de la matrice représentant l'ensemble de la population, nous sélectionnerons un agent et son voisinage pour créer une liste de 2 à 9 personnes : la **liste discussion** (le nombre de personnes choisies parmi

ses voisins se fait aléatoirement). Si un individu se situe au bord de la matrice, on prend également dans la liste les agents qui seraient ses voisins si on joignait les bords et les coins de la matrice. *codé par Ruxue et Aya*

S'il n'y a que 2 personnes dans la liste discussion, seule la **force de persuasion** influera et ce sur un seul agent, celui dont la force de persuasion est la plus faible. La mise à jour des convictions est alors régie par la loi suivante : lorsque  $p_i < fp_j$ , on applique la formule  $p_i \leftarrow p_i + fp_j * (p_j - p_i)$  *codé par Ruxue et Jenny*

Si la discussion comprend de 3 à 9 personnes, ce sera le **voisinage** qui influera sur la conviction de chacun. Chaque individu modifie sa conviction en fonction de la moyenne des convictions des autres pondérée de leur force de persuasion ( $moy = \frac{\sum p_i * fp_i}{\sum fp_i}$ ) et du seuil de persuasion  $\sigma$  : si  $|p_i - moy| < \sigma$  alors  $p_i \leftarrow moy$  sinon,  $p_i$  ne change pas. Nous estimons que seul le voisinage influera dans ce cadre. Comme le corroborent certaines expériences telles que celle menée par Asch dans les années 1950 par exemple (explication : <https://sciencetonante.wordpress.com/2013/04/22/l'experience-de-asch-sur-le-conformisme/>) l'humain tend à se conformer au comportement d'autrui lorsqu'il est entouré. C'est pourquoi nous ne prenons en compte que les convictions des autres et pas la crédibilité de la fake news (qui entrera en jeu plus tard).

La fonction qui modifie ainsi les convictions des agents de la population est nommée *influ\_voisinage*. *codé par Aya*

Ensuite, nous avons codé la partie "Interaction à distance" c'est-à-dire la transmission d'une fake news par le biais d'un réseau social. Des personnes physiquement éloignées pourront alors interagir. Comme évoqué dans la partie "Initialisation du système", le réseau virtuel nous fournit, pour chaque agent, la liste des autres agents qui voient la news si celui-ci la partage (on peut dire que c'est sa liste d'amis ou d'abonnés). Pour faire un parallèle avec la partie "Discussion physique", on peut considérer que c'est l'équivalent de la liste discussion. Dans un premier temps, un média malveillant est choisi au hasard. Son opinion est alors mise à 1 et il partage la news à sa liste d'abonnés grâce à l'algorithme décrit ci-dessous : c'est le lancement de la fake news.

Pour ce qui est du partage de l'information, le processus est semblable à celui de la discussion physique à quelques différences près. On met toujours à jour les convictions selon plusieurs facteurs :

- la **force de persuasion** jouera le même rôle que précédemment. Nous estimons que même à travers un réseau social ou un article, la force de persuasion a son rôle à jouer.

- contrairement à la discussion physique où la conviction dépend exclusivement du voisinage, lorsqu'un agent se retrouve seul derrière son écran ou son journal, le rôle de la **crédibilité de la news** que nous nommerons "*news*" est prépondérant (on notera au passage que cette valeur (comprise entre 0 et 1) est la seule caractéristique propre à la news). La conviction  $p$  de l'agent de force de persuasion  $fp$  sera modifiée selon la loi suivante :  $p \leftarrow p + (1 - fp) * (news - p)$ . En effet, plus sa force de persuasion sera faible, plus il sera enclin à croire à la news et sa conviction tendra alors plus fortement vers la valeur de la crédibilité de la news.

La fonction que nous venons de décrire se nomme *influ\_partage*. *codé par Aya*

(*Rédaction : Aya – relecture : Jenny*)

- Simulation :

La dernière étape de la réalisation de notre modèle étant la simulation, cette étape nous permet de visualiser la variation des opinions et des convictions, d'avoir une première idée sur notre modèle. Est-ce qu'elle est cohérente ? Correspond-t-elle à ce que nous attendions ?

Nous avons donc fait en total 6 simulations pour voir la variation des opinions et des convictions dans différents environnements. Toutes les simulations sont basées sur les mêmes principes. Il s'agit de créer une matrice qui mémorise tous les opinions ou convictions de la population après chaque étape jusqu'au nombre d'étapes donné.

Dans la simulation finale, à chaque étape, un individu et un mode de transmission seront choisis aléatoirement. Si c'est par discussion physique, la liste discussion est créée (en veillant à retirer les médias de la liste) et les convictions sont changées par la fonction *influ\_voisinage*. Si c'est à distance, les convictions sont modifiées par la fonction *influ\_partage*. Ensuite, la fonction *simulation* va parcourir tous les individus de la population, si leur conviction est supérieure à 0.5 et ils ne croient pas au news, alors ils y croiraient ; contrairement si leur conviction est inférieure à 0.5 et ils ne croient pas, alors ils ne croiraient plus. Une matrice va être créée ensuite pour mémoriser l'opinion ou la conviction de chaque individu en parcourant la population modifiée. Et toutes les matrices seront enregistrées par la liste finale. *codé par Ruxue*

Enfin, une fonction animation permet de montrer la variation en affichant successivement toutes les matrices de la population après chaque étape. *codé par Ruxue*

(*Rédaction* : Ruxue – *relecture* : Aya)

- Analyse : *codé par Jenny*

Nous avons choisi de représenter, à l'aide de graphiques, l'effet de nos paramètres sur nos différents indicateurs. Ces paramètres sont la **connectivité** du milieu, la **crédibilité** de la news et le degré de **malveillance** de l'environnement, défini comme le ratio du nombre d'organismes médiatiques malveillants sur le nombre total d'organismes médiatiques. Les indicateurs, quant à eux, peuvent être définis comme des éléments montrant l'évolution de notre modèle. On compte parmi ceux-ci le **temps d'adoption** de la news, c'est-à-dire le nombre d'étapes afin que 90% de la population ait son opinion à 1, le **taux d'opinion à 1** à la dernière étape et enfin, la **moyenne des convictions** finales. Certaines expériences n'étant pas vraiment pertinentes, nous n'avons pas représenté l'influence de tous les paramètres en fonction de tous les indicateurs mais uniquement la moyenne des convictions par rapport à la crédibilité de la news (*MOY(news)*), le taux d'adoption en fonction de la crédibilité de la news (*TA(news)*), le temps d'adoption en fonction de la connectivité du milieu (*TPS(co)*) et enfin, le temps d'adoption en fonction du degré de malveillance du milieu (*TPS(malv)*).

Dans la représentation *MOY(news)*, on peut remarquer que plus la news est crédible, plus la moyenne des convictions s'approche de 1 c'est-à-dire plus le nombre de personnes convaincus dans la population est important, ce qui a du sens.

Dans la représentation *TA(news)*, on peut observer que plus la news est crédible, plus le taux de personnes convaincus par la news est élevé ; cela est cohérent avec la remarque faite précédemment.

Dans la représentation *TPS(co)*, on peut noter que plus un milieu est connecté, plus le nombre d'étapes nécessaire pour que 90% de la population croie en la news est faible : en effet, une haute connectivité du milieu permet une rapide propagation d'une information et donc, une évolution rapide des opinions des agents.

Enfin, dans la représentation *TPS(malv)*, on peut voir que plus le nombre d'organismes malveillants présents dans la population est élevée, plus le nombre d'étapes pour observer une population majoritairement convaincu est faible. Cela est dû au fait que les organismes malveillants ont un poids plus important dans la population : ils ont une importante force de persuasion et une opinion maximale.

(Rédaction : Jenny – relecture : Ruxue)

## Conclusion :

Nous croyons que les démarches mises en œuvre dans le cadre de ce projet s'approche de près du travail d'un ingénieur : réalisation d'un cahier des charges, respect du délais imparti, travail en équipe, prise de conscience. Ces semaines de travail ont donc été très utiles et le seront davantage pour notre future carrière.

De plus, ce projet nous aura permis d'acquérir de nos nouvelles compétences : notamment, l'utilisation dans un cas très concret du langage Python.

La nature de l'environnement dans lequel la population se trouve, s'il est connecté ou non, va influencer le temps de propagation. Cependant, le résultat final reste le même : dans le sens où, à l'infini, soit tout le monde va croire à la news ou alors personne ne va y croire. Quand une personne est exposée à une information, elle va chercher l'opinion des autres et selon leur force de persuasion et/ou la crédibilité de la news, elle peut choisir de croire ou ne pas croire.

Nous avons rencontrés des problèmes durant ce projet. Alors, nous avons réfléchi ensemble, demandé de l'aide aux professeurs pour les résoudre.

Enfin, nous devons reconnaître que notre modèle présente toujours quelques incompatibilités: en effet, l'état de notre modèle à l'infini, mentionné ci-dessus, ne correspond pas à la réalité. Dans la réalité, même à l'infini, il est possible d'observer des opinions diversifiées dans une population plus ou moins grande. Nous pouvons donc toujours améliorer notre dynamique de propagation en précisant encore certaines conditions et en modifiant certaines formules.

(Rédaction : Ruxue – relecture : Jenny)

## Résumé en anglais :

Our project was all about the modelling and the study of the propagation of a fake news in an environment where you can find random human beings but also media organizations. Our aim was to highlight the influence of different parameters both on the extent of which the news propagates itself and on the speed of its propagation.

In order to achieve our objective, agents have been placed on a virtual two-dimensional grate and can interact with one another either physically or from afar through social networks, if they are friends or so.

The outcome can be considered as final. We will come to realize, for example, that a fake news propagates itself much faster when the average number of social networks on which an agent is active increases.

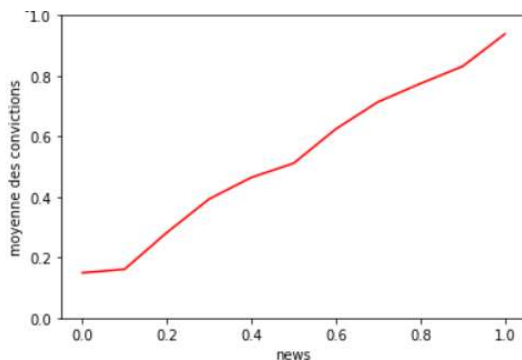
(Rédaction : Jenny – relecture : Aya)

## Annexes :

### - Graphiques :

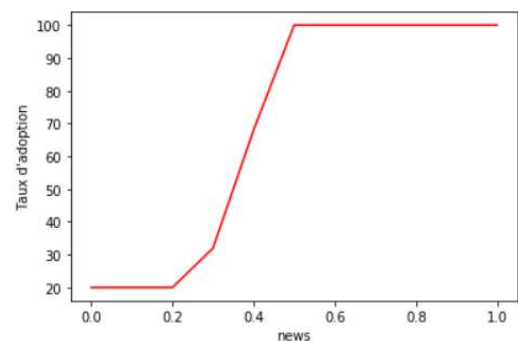
#### ➤ MOY en fonction de NEWS :

Moyenne des convictions en fonction de la crédibilité de la fake news



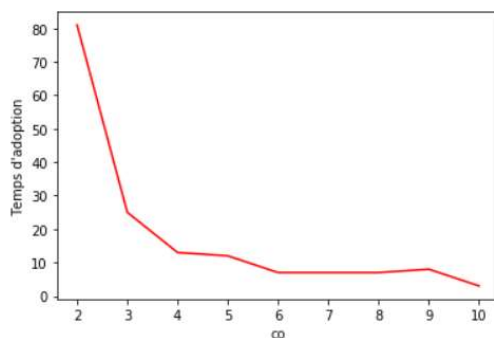
#### ➤ TA en fonction de NEWS :

Taux d'opinions à 1 à la fin de la simulation en fonction de la crédibilité de la fake news



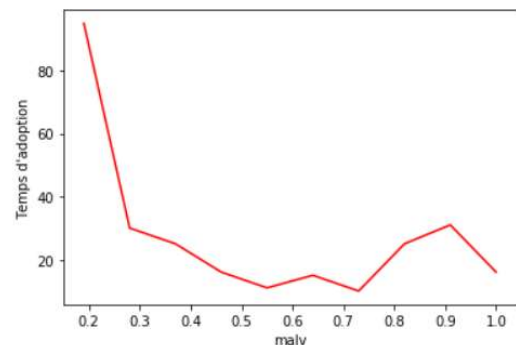
#### ➤ TPS en fonction de CO :

Nombre d'étapes nécessaires pour que 90% de la population croie en la news en fonction de co



#### ➤ TPS en fonction de MALV :

Nombre d'étapes nécessaires pour que 90% de la population croie en la news en fonction de malv





## - Spécification du modèle :

### Attributs des agents

#### *Etres humains*

Nom	Type	Intervalle	Valeur initiale	Fixe ?	Explication
nom	str	[str(0),str(N)]	str(k)	Oui	Nom : nom de l'agent (clé du dict)
op	int	{0,1}	0	Non	Opinion : y croit (1) ou non (0)
deja_vu	int	[0,steps]	0	Non	Déjà vu : nombre d'expositions à l'information par le biais de quelqu'un qui y croit
fp	float	[0,1]	Random	Oui	Force de persuasion : capacité à convaincre quelqu'un (accrue qd tend vers 1)
p	float	[0,1]	0.5	Non	Conviction : caractérise la conviction à la news (tend vers 1) ou non (tend vers 0). Si assez élevée, fait basculer l'opinion de 0 à 1 et inversement

#### *Organismes médiatiques*

Nom	Type	Intervalle	Valeur initiale	Fixe ?	Explication
nom	str	['OM'+str(0),'OM'+str(N)]	'OM'+str(k)	Oui	Nom : nom de l'organisme (clé du dict)
op	int	{0,1}	0	Non	Opinion : ne relaie l'info que s'il l'estime correcte (1)
deja_vu	int	[0,steps]	0	Non	Déjà vu : nombre d'expositions à l'info par le biais de quelqu'un qui y croit
fp	float	[0,1]	Random mais >0.7	Oui	Force de persuasion : accrue car organisme médiatique
p	float	[0,1]	0.5	Non	Conviction :

#### *Organismes médiatiques malveillants*

Nom	Type	Intervalle	Valeur initiale	Fixe ?	Explication
nom	str	['XOM'+str(0),'XOM'+str(N)]	'XOM'+str(k)	Oui	Nom : nom de l'organisme (clé du dict)
op	int	{0,1}	0	Non	Opinion : relaie l'info même si n'y croit
deja_vu	int	[0,steps]	0	Non	Déjà vu : nombre d'expositions à l'info par le biais de quelqu'un qui y croit
fp	float	[0,1]	Random mais >0.7	Oui	Force de persuasion : accrue car organisme médiatique
p	float	[0,1]	0.75	Oui	Conviction : toujours à 0.75 car cherche à garder le p des autres au dessus de 0.5

### Paramètres du modèle :

Nom	Type	Intervalle	Valeur initiale	Fixe ?	Explication
news	float	[0,1]	à entrer	Oui	Crédibilité de la news : crédible si tend vers 1
malv	float	[0,1]	à entrer	Oui	Degré de malveillance de l'environnement : proportion d'agents malveillants parmi les médias
co	int	range(11)	à entrer	Oui	Connectivité du milieu : nombre de réseaux sociaux moyen par individu

## Indicateurs :

Nom	Type	Intervalle	Explication
MOY	float	[0,1]	Moyenne des convictions de tous les agents à la fin d'une simulation
TA	float	[0,100]	Taux d'adoption : pourcentage d'opinions à 1 à la fin d'une simulation
TPS	int	range(steps)	Temps d'adoption : nombre d'étapes qu'il faut pour que le pourcentage d'opinions à 1 soit au moins égal à 90%

## Liste des expériences :

Ci-dessous la liste des expériences qui nous ont semblé les plus pertinentes.

MOY en fonction de news

TA en fonction de news

TPS en fonction de co

TPS en fonction de malv

## - Fonctions :

```
def simu_p_finale(pop, res_virtu, names, n_1dim, steps, list_xom, mat_pop, news):  
    pop_ap_lan, lanceur = lancement_fake_news(pop, list_xom)  
    results = [matrice_convic(pop, names, n_1dim)]  
  
    for simu in range(steps):  
        if simu == 0:  
            i = lanceur  
            Res_p_changé = influ_partage(pop, res_virtu, i, news)  
        else:  
            i = np.random.choice(names)  
            k = np.random.choice([0, 1])  
            if k == 0:  
                list_discu = selection_voisinage(mat_pop, n_1dim, i)  
                for a in list_discu:  
                    if '0' in a:  
                        list_discu = list_discu[list_discu != a]  
                if len(list_discu) >= 2:  
                    Res_p_changé = influ_voisinage(list_discu, pop, seuil)  
            else:  
                Res_p_changé = influ_partage(pop, res_virtu, i, news)  
  
        for c in Res_p_changé:  
            op, deja, fp, p = Res_p_changé[c]  
            if p > 0.5 and op == 0:  
                op = 1  
                Res_p_changé[c] = (op, deja, fp, p)  
            elif p < 0.5 and op == 1:  
                op = 0  
                Res_p_changé[c] = (op, deja, fp, p)  
  
        P = []  
        for e in mat_pop.ravel():  
            op, deja, fp, p = Res_p_changé[e]  
            P.append(p)  
        P_final = np.reshape(P, (n_1dim, n_1dim))  
        results.append(P_final.copy())  
    return results  
  
Simu_p_finale = simu_p_finale(Pop_ac_om, Reseau_virtuel, Noms_agents_avec, N_1dim, steps, List_xom, Mat_pop_ac_om, News)  
Simu_p_finale
```