

# Simulation individu-centrée de la propagation d'une maladie infectieuse et évaluation d'une stratégie de dépistage et vaccination

Valentina Hammel-Monzon (20274033)

 [valentina9000](#)

Mathilde Nguyen (20267325)

 [Mathilde389](#)

Travail présenté à Timothée Poisot dans le cadre du cours  
BIO 2045 « Simuler le Vivant »

[Source du projet](#)

2026-04-14

# Table des matières

- Introduction ..... 1
- Présentation du modèle ..... 1
- Implémentation ..... 2
  - Packages nécessaires ..... 2
  - Création des types ..... 2
  - Génération d'agents aléatoires ..... 3
  - Fonctions utiles ..... 4
  - Initialisation ..... 4
  - Paramètres intervention ..... 4
  - Simulation principale ..... 5
  - Analyse des résultats ..... 7
  - Graphique comparatif ..... 7
  - Discussion ..... 10
- Bibliographie ..... 11

## Introduction

Lors de propagation rapide de maladies infectieuses, comme les épidémies, il est essentiel de comprendre comment différentes stratégies d'intervention permettent de limiter la transmission et la mortalité. En effet, cela représente un enjeu majeur en santé publique, surtout lorsque les individus infectés peuvent transmettre la maladie sans présenter de symptômes. Par exemple, dans le cas de la covid, le contrôle de la propagation de la maladie a été compliqué du aux individus porteurs asymptomatiques (Zhang et al, 2021).

Afin de contrôler les épidémies, il est important de mettre en place des outils comme le dépistage et la vaccination. Les tests de dépistages, comme les tests antigéniques rapides, permettent d'identifier rapidement les individus infectieux (Mina et al, 2020). Les individus infectieux peuvent alors prendre des mesures de sécurités afin de pas infecter le reste de la population. Les vaccins sont des mesures de prévention qui permettent de donner de l'immunité protectrice comme celui pour l'hépatite B (Schillie et al, 2018). Cela empêche le développement de la maladie lors d'un contact par la suite.

Cependant l'efficacité de ces stratégies dépend de plusieurs contraintes biologiques et logistiques. Les tests de dépistage ne sont pas totalement fiable et peuvent donner des faux positives ou des faux négatifs. Les faux négatifs sont ceux qui mettent à risque le reste de la population puisqu'un individu infecté est identifié comme sain et peut donc infecter d'autres personnes. De plus les vaccins n'offrent pas toujours une protection immédiate, et il y a donc une période pendant laquelle les individus vaccinés sont vulnérable. Le budget attribué aux stratégies de vaccination est également une limite.

L'objectif de ce travail est donc de tester une stratégie de contrôle d'une maladie infectieuse dans une population en agissant sur le dépistage et la vaccination des individus.

## Présentation du modèle

Nous utilisons un modèle de simulation individu-centré pour représenter la propagation d'une maladie infectieuse dans une population. Les modèles individu-centrés permettent de représenter explicitement les interactions entre individus et d'évaluer l'effet de différentes stratégies de contrôle, notamment le dépistage et la vaccination (Adam & Arduin, 2023). Chaque individu est représenté par un agent évoluant dans un espace discret bidimensionnel (lattice) défini par des coordonnées allant de  $-50$  à  $50$  sur les axes  $x$  et  $y$ . La population initiale est composée de 3750 agents distribués aléatoirement dans cet espace.

Chaque agent possède plusieurs caractéristiques : une position spatiale, un état sanitaire (infectieux ou sain), un compteur de durée d'infection (fixé à 21 jours), ainsi qu'un identifiant unique. Lorsqu'un agent devient infectieux, il le reste pendant toute la durée de la maladie, qui est toujours fatale dans ce modèle. La population est initialement entièrement naïve, c'est-à-dire qu'aucun individu ne possède d'immunité au départ.

À chaque pas de temps (génération), les agents se déplacent aléatoirement dans la grille. La transmission de la maladie se produit lorsqu'un agent infectieux partage la même cellule qu'un agent sain. Dans ce cas, la probabilité de transmission est fixée à  $0,4$ . Les agents infectieux voient leur durée d'infection diminuer à chaque génération, et sont retirés de la population lorsque cette durée atteint zéro, ce qui correspond à leur décès.

Afin de simuler une intervention sanitaire, un mécanisme combinant dépistage et vaccination a été intégré au modèle. L'intervention est déclenchée uniquement après la détection du premier

décès dans la population. Une stratégie de dépistage actif est alors mise en place à l'aide de tests antigéniques rapides (RAT), qui permettent d'identifier les individus infectieux avec une probabilité élevée (Lasser et al., 2021 ; Tran et al., 2023). Lorsqu'un cas est détecté, une intervention ciblée est appliquée à ses contacts, ce qui reflète les approches modernes de contrôle en épidémiologie (Lasser et al., 2021).

Dans ce modèle, les tests RAT présentent une probabilité de détection de 95 %, pour un coût unitaire de 4 \$, avec un budget total d'intervention fixé à 21 000 \$. Ils offrent donc un moyen rapide d'identifier les individus infectieux, bien que leur sensibilité imparfaite puisse en limiter l'efficacité dans le contrôle global de l'épidémie (Lasser et al., 2021).

Lorsqu'un individu est testé positif, les agents présents dans la même cellule spatiale (considérés comme ses contacts proches) peuvent être vaccinés, à condition que le budget le permette. Le coût d'une vaccination est de 17\$ par individu. Le vaccin est entièrement efficace, mais son effet n'est actif qu'après un délai de deux générations. Une fois le vaccin actif, l'agent ne peut plus être infecté ni transmettre la maladie.

La simulation se poursuit jusqu'à l'extinction de la maladie (absence d'individus infectieux) ou jusqu'à un maximum de 2000 générations. Les principales variables suivies sont le nombre d'individus sains et infectieux au cours du temps, ainsi que le nombre total de décès, permettant d'évaluer l'efficacité de la stratégie de vaccination en comparaison avec un scénario sans intervention.

## Implémentation

### Packages nécessaires

```
# Initialisation

using Random
using CairoMakie
using UUIDs

Random.seed!(123456)
CairoMakie.activate!(px_per_unit=6.0)

# Identifiants uniques pour chaque agent

UUIDs.uuid4()
```

```
Base.UUID("0e7f03b4-5348-4033-b84e-99016a03d103")
```

### Création des types

```
# Structure d'un agent (individu)

Base.@kwdef mutable struct Agent
    x::Int64 = 0          ## position x
    y::Int64 = 0          ## position y
    clock::Int64 = 21     ## durée de la maladie est de 21 jours
    infectious::Bool = false ## état infectieux
    tested::Bool = false
    # Vaccination
    vaccinated::Bool = false ## vacciné ou non
```

```

    days_after_vax::Int64 = 0      ## délai avant efficacité
    id::UUIDs.UUID = UUIDs.uuid4() ## identifiant unique
end

# Structure du paysage (espace de simulation)

Base.@kwdef mutable struct Landscape
    xmin::Int64 = -50
    xmax::Int64 = 50
    ymin::Int64 = -50
    ymax::Int64 = 50
end

# Création du paysage

L = Landscape(xmin=-50, xmax=50, ymin=-50, ymax=50)

```

```
Main.var"##277".Landscape(-50, 50, -50, 50)
```

## Génération d'agents aléatoires

```

# Crée un agent à une position aléatoire
Random.rand(::Type{Agent}, L::Landscape) = Agent(x=rand(L.xmin:L.xmax), y=rand(L.ymin:L.ymax))

# Crée plusieurs agents
Random.rand(::Type{Agent}, L::Landscape, n::Int64) = [rand(Agent, L) for _ in 1:n]

# Cette fonction nous permet donc de générer un nouvel agent dans un paysage:
rand(Agent, L)

# Mais aussi de générer plusieurs agents:
rand(Agent, L, 3)

# Déplacement des agents

function move!(A::Agent, L::Landscape; torus=true)
    A.x += rand(-1:1)
    A.y += rand(-1:1)
    if torus
        A.y = A.y < L.ymin ? L.ymax : A.y
        A.x = A.x < L.xmin ? L.xmax : A.x
        A.y = A.y > L.ymax ? L.ymin : A.y
        A.x = A.x > L.xmax ? L.xmin : A.x
    else
        A.y = A.y < L.ymin ? L.ymin : A.y
        A.x = A.x < L.xmin ? L.xmin : A.x
        A.y = A.y > L.ymax ? L.ymax : A.y
        A.x = A.x > L.xmax ? L.xmax : A.x
    end
    return A
end
end

```

```
move! (generic function with 1 method)
```

## Fonctions utiles

```
# Déterminer si l'agent est infectueux
isinfectious(agent::Agent) = agent.infectious

# Déterminer si l'agent est sain
ishealthy(agent::Agent) = !isinfectious(agent)

# Type population = liste d'agents
const Population = Vector{Agent}

# Filtrage des agents
infectious(pop::Population) = filter(isinfectious, pop)
healthy(pop::Population) = filter(ishealthy, pop)

# Agents dans la même cellule (même position)
incell(target::Agent, pop::Population) = filter(ag -> (ag.x, ag.y) == (target.x, target.y),
pop)
```

```
incell (generic function with 1 method)
```

## Initialisation

```
# Génération de la population
function make_population(L::Landscape, n::Int)
    return rand(Agent, L, n)
end

# Simplification de l'affichage de cette population
Base.show(io::IO, ::MIME"text/plain", p::Population) = print(io, "Une population avec
$(length(p)) agents")

# Génération de la population initiale
population = make_population(L, 3750)

# Cas index (premier infecté)
rand(population).infectious = true
```

```
true
```

## Paramètres intervention

```
budget = 21000      ## budget total
cost_vaccine = 17  ## coût vacciné
cost_test = 4       ## coût test

intervention_started = false ## commence après premier décès
first_death = false
deaths = 0          ## compteur de morts
```

```
0
```

## Fonction de vaccination

```
function vaccinate!(agent::Agent)
    if !agent.vaccinated
        agent.vaccinated = true
        agent.days_after_vax = 0    ## délai avant activation = 2 jours (immunité après 2
    ticks)
    end
end
```

vaccinate! (generic function with 1 method)

## Simulation principale

```
Base.@kwdef struct InfectionEvent
    tick::Int
    from::UUIDs.UUID
    to::UUIDs.UUID
    x::Int
    y::Int
end

function run_simulation(L::Landscape, n::Int, budget_total; with_intervention=true)

    population = make_population(L, n)
    rand(population).infectious = true

    events = InfectionEvent[]

    budget = budget_total
    intervention_started = false
    first_death = false
    deaths = 0

    tick = 0
    maxlength = 2000
    S = Int[]
    I = Int[]
    D = Int[]

    while (length(infectious(population)) > 0) && (tick < maxlength)

        tick += 1

        # Déplacement
        for agent in population
            move!(agent, L)
        end

        # Mise à jour délai vaccin
        for agent in population
            if agent.vaccinated
                agent.days_after_vax += 1
            end
        end

        # Infection
        for agent in Random.shuffle(infectious(population))
            for neighbor in healthy(incell(agent, population))

                immune = neighbor.vaccinated && neighbor.days_after_vax >= 2
```

```

        if rand() <= 0.4 && !immune
            neighbor.infectious = true
            push!(events, InfectionEvent(tick, agent.id, neighbor.id, neighbor.x,
neighbor.y))
        end
    end
end

# Progression maladie
for agent in infectious(population)
    agent.clock -= 1
end

# Décès
before = length(population)
population = filter(a -> a.clock > 0, population)
after = length(population)

# Déclenche intervention
if !first_death && after < before
    intervention_started = true
    first_death = true
end

deaths += (before - after)

# Intervention (activable)
if with_intervention && intervention_started && budget > 0

    for agent in population

        if !agent.tested && budget >= 4
            budget -= 4
            agent.tested = true

            positive = agent.infectious ? rand() <= 0.95 : rand() <= 0.05

            if positive
                for neighbor in incell(agent, population)
                    if budget >= 17 && !neighbor.vaccinated
                        vaccinate!(neighbor)
                        budget -= 17
                    end
                end
            end
        end

        if budget <= 0
            break
        end
    end
end

# Suivi des états
push!(S, length(healthy(population)))
push!(I, length(infectious(population)))
push!(D, deaths)

end

return deaths, budget, events, S, I, D
end

```

```
run_simulation (generic function with 1 method)
```

## Analyse des résultats

```
using Statistics

n_runs = 50

results_with = [run_simulation(L, 3750, 21000; with_intervention=true) for _ in 1:n_runs]
results_without = [run_simulation(L, 3750, 21000; with_intervention=false) for _ in 1:n_runs]

# Une simulation pour visualisation (courbes temporelles)
sim_with = run_simulation(L, 3750, 21000; with_intervention=true)
sim_without = run_simulation(L, 3750, 21000; with_intervention=false)

S_with, I_with, D_with = sim_with[4], sim_with[5], sim_with[6]
S_without, I_without, D_without = sim_without[4], sim_without[5], sim_without[6]

deaths_with = [r[1] for r in results_with]
budget_with = [r[2] for r in results_with]

deaths_without = [r[1] for r in results_without]
budget_without = [r[2] for r in results_without]

println("Budget moyen restant (avec intervention) = ", mean(budget_with))

mean_with = mean(deaths_with)
mean_without = mean(deaths_without)

var_with = var(deaths_with)
var_without = var(deaths_without)

println("\n RÉSULTATS SUR ", n_runs, " SIMULATIONS")

println("\n AVEC intervention :")
println("Moyenne des morts = ", mean_with)
println("Variance = ", var_with)

println("\n SANS intervention :")
println("Moyenne des morts = ", mean_without)
println("Variance = ", var_without)

println("\n GAIN (morts évités) = ", mean_without - mean_with)
```

Budget moyen restant (avec intervention) = 1578.28

RÉSULTATS SUR 50 SIMULATIONS

AVEC intervention :

Moyenne des morts = 2425.32

Variance = 389024.0587755102

SANS intervention :

Moyenne des morts = 2739.9

Variance = 669576.9489795918

GAIN (morts évités) = 314.57999999999999

## Graphique comparatif

```
f1 = Figure();

ax = Axis(f1[1, 1];
    xlabel="Simulation",
    ylabel="Nombre de morts",
```

```

    title="Comparaison des décès avec et sans intervention")

scatter!(ax, 1:length(deaths_with), deaths_with, label="Avec intervention", color=:blue)
scatter!(ax, 1:length(deaths_without), deaths_without, label="Sans intervention", color=:red)

axislegend(ax)
save("sans-intervention".png, f1)
display(f1)

```

```

ERROR: FieldError: type String has no field `png`; String has no fields at all.

```

**Figure 1: Comparaison des décès** Cette figure présente, pour chacune des 50 simulations, le nombre total de décès observés avec et sans intervention. Chaque point correspond à une simulation indépendante. Cette figure permet de visualiser la variabilité des résultats entre les répétitions et de comparer directement l'effet global de l'intervention sur la mortalité. Elle met en évidence la dispersion des résultats ainsi que la tendance générale à une réduction du nombre de décès en présence d'une intervention.

```

println("Moyenne morts avec intervention = ", mean(deaths_with))
println("Moyenne morts sans intervention = ", mean(deaths_without))

f2 = Figure();

# Graphique avec intervention
ax1 = Axis(f2[1, 1],
    title = "Évolution AVEC intervention",
    xlabel = "Temps",
    ylabel = "Population")

lines!(ax1, 1:length(S_with), S_with, label="Sains")
lines!(ax1, 1:length(I_with), I_with, label="Infectieux")
lines!(ax1, 1:length(D_with), D_with, label="Décédés")

axislegend(ax1)

# Graphique sans intervention
ax2 = Axis(f2[2, 1],
    title = "Évolution SANS intervention",
    xlabel = "Temps",
    ylabel = "Population")

lines!(ax2, 1:length(S_without), S_without, label="Sains")
lines!(ax2, 1:length(I_without), I_without, label="Infectieux")
lines!(ax2, 1:length(D_without), D_without, label="Décédés")

axislegend(ax2)
save("evolution".png, f2)
display(f2)

```

```

Moyenne morts avec intervention = 2425.32
Moyenne morts sans intervention = 2739.9

```

```

ERROR: FieldError: type String has no field `png`; String has no fields at all.

```

**Figure 2: Dynamique temporelle de l'épidémie avec intervention** Cette figure illustre l'évolution du nombre d'individus sains, infectieux et décédés au cours du temps pour une simulation représentative avec intervention. Elle permet de suivre la progression de l'épidémie génération par génération, d'identifier le pic d'infection et d'observer l'effet de l'intervention sur la diminution du nombre d'individus infectieux et la limitation des décès.

**Figure 3: Dynamique temporelle de l'épidémie sans intervention** Cette figure présente l'évolution du nombre d'individus sains, infectieux et décédés au cours du temps pour une simulation sans intervention. Elle sert de référence pour comparer la dynamique naturelle de l'épidémie en absence de contrôle. Elle permet notamment d'observer une propagation plus rapide de la maladie, un pic d'infection plus élevé et une augmentation plus importante du nombre de décès.

Les Figures 2 et 3 permettent une comparaison directe de la dynamique temporelle de l'épidémie entre les deux scénarios, tandis que la Figure 1 synthétise l'effet global de l'intervention sur plusieurs simulations. Ensemble, ces figures permettent d'évaluer à la fois l'effet moyen de l'intervention et la variabilité des trajectoires épidémiques.

Les résultats du modèle sont obtenus à partir de 50 simulations indépendantes réalisées avec et sans intervention. Cette réplication permet de tenir compte de la variabilité inhérente au caractère stochastique du modèle.

En moyenne, le nombre de décès observé en présence d'une intervention est de 2425.32 individus, alors qu'il atteint 2739.9 individus en absence d'intervention. Cela correspond à une réduction moyenne d'environ 315 décès lorsque le dépistage et la vaccination sont appliqués. Ainsi, l'intervention permet de diminuer la mortalité globale au sein de la population simulée.

La variabilité des résultats diffère également entre les deux scénarios. La variance du nombre de décès est de 389024.06 avec intervention, contre 669576.95 sans intervention. Les simulations avec intervention présentent donc une dispersion plus faible, indiquant des résultats plus homogènes d'une répétition à l'autre.

Le budget moyen restant à la fin des simulations avec intervention est de 1578.28, ce qui indique que la majorité des ressources disponibles est utilisée au cours de l'épidémie. En revanche, dans le scénario sans intervention, le budget demeure constant, puisqu'aucune dépense n'est effectuée.

Les dynamiques temporelles de l'épidémie sont illustrées par les courbes représentant l'évolution du nombre d'individus sains, infectieux et décédés. Sans intervention, le nombre d'individus infectieux augmente rapidement, entraînant une diminution rapide du nombre d'individus sains et une accumulation importante de décès. Avec intervention, l'augmentation du nombre d'infectieux est plus progressive, et la diminution des individus sains est ralentie, ce qui se traduit par un nombre total de décès plus faible.

L'analyse des événements d'infection révèle un total de 121 216 transmissions enregistrées au cours des simulations avec intervention. De plus, 73 344 agents infectieux uniques sont impliqués dans ces événements. La distribution du nombre d'infections causées par individu montre que la majorité des agents infectent un petit nombre d'individus, tandis qu'une minorité contribue à un nombre plus élevé de transmissions.

Dans l'ensemble, ces résultats mettent en évidence l'impact mesurable de l'intervention sur la réduction de la mortalité et sur la dynamique de propagation de la maladie, tout en illustrant la variabilité importante des trajectoires épidémiques entre simulations.

## Discussion

Les résultats obtenus montrent que l'intervention basée sur le dépistage et la vaccination ciblée permet de réduire le nombre total de décès par rapport au scénario sans intervention. En moyenne, environ 314 décès sont évités, ce qui indique que la stratégie mise en place est efficace pour limiter la propagation de la maladie. Cette réduction s'explique par le fait que le dépistage permet d'identifier rapidement une partie des individus infectieux et d'interrompre les chaînes de transmission, tandis que la vaccination des contacts proches diminue la probabilité de transmission locale (Adam & Arduin, 2023).

La stratégie de vaccination implémentée correspond à une approche de vaccination en anneau, dans laquelle les individus en contact avec un cas détecté sont vaccinés de manière ciblée (Henao-Restrepo et al., 2017). Cette stratégie vise à créer une barrière immunitaire autour des foyers d'infection afin de limiter leur expansion (Krauland et al., 2026). Dans le modèle, cette approche permet de contenir localement la propagation, mais son efficacité dépend fortement de la rapidité du dépistage et de la capacité à intervenir avant que l'infection ne se diffuse à d'autres cellules.

Cependant, l'intervention ne permet pas d'éliminer complètement l'épidémie. Plusieurs facteurs expliquent cette efficacité partielle. D'une part, les tests antigéniques ne sont pas parfaitement sensibles, ce qui signifie qu'une fraction des individus infectieux n'est pas détectée et continue de transmettre la maladie (Adam & Arduin, 2023 ; Krauland et al., 2026). D'autre part, le vaccin n'est pas immédiatement efficace; le délai de deux générations avant l'acquisition de l'immunité laisse une période durant laquelle les individus vaccinés restent susceptibles. Enfin, la limitation du budget restreint le nombre total de tests et de vaccinations pouvant être réalisés, ce qui empêche une couverture complète de la population.

L'analyse du budget apporte un éclairage important sur l'efficacité de l'intervention. En moyenne, une partie du budget reste inutilisée à la fin des simulations, ce qui suggère que l'épidémie peut parfois s'éteindre avant que toutes les ressources ne soient déployées. Cela indique que le facteur limitant n'est pas uniquement la quantité de ressources disponibles, mais également le moment du déclenchement de l'intervention et la dynamique de propagation de la maladie. Toutefois, dans certaines simulations, une consommation plus rapide du budget limite la capacité à tester et vacciner de nouveaux individus au cours de l'épidémie. Cela met en évidence l'importance d'une allocation efficace et précoce des ressources pour maximiser l'impact des interventions.

Les résultats montrent également une forte variabilité entre les simulations, tant avec que sans intervention. Cette variabilité est liée au caractère stochastique du modèle, notamment dans les déplacements des agents et les événements de transmission (Adam & Arduin, 2023 ; Shakiba et al., 2021). Dans certains cas, l'épidémie s'éteint rapidement après peu de transmissions, tandis que dans d'autres, elle se propage largement dans la population. Cette variabilité reflète l'importance du hasard dans les dynamiques épidémiques, en particulier au début de l'épidémie (Shakiba et al., 2021 ; Karen et al., 2022).

D'un point de vue épidémiologique, ces résultats sont cohérents avec les observations réelles. Ils soulignent l'importance des stratégies combinant dépistage et vaccination ciblée pour

contrôler une épidémie (Adam & Arduin, 2023). Le modèle met également en évidence le rôle crucial du délai d'action des interventions; une réponse tardive ou une immunité retardée peut limiter l'efficacité globale des mesures (Krauland et al., 2026). Par ailleurs, la distribution du nombre d'infections secondaires suggère une propagation hétérogène, où certains individus infectent plusieurs autres, tandis que la majorité en infecte peu. Ce phénomène est caractéristique des dynamiques de type « super-propagation », soit des situations où un petit nombre d'individus infectés est responsable d'un nombre disproportionné de transmissions, contribuant fortement à la diffusion globale de la maladie (Karen et al., 2022 ; Shakiba et al., 2021).

On peut aussi citer des références dans le document `references.bib`, qui doit être au format BibTeX. Les références peuvent être citées dans le texte avec @ suivi de la clé de citation. Par exemple: [1] – la bibliographie sera ajoutée automatiquement à la fin du document.

Le format de la bibliographie est American Physics Society, et les références seront correctement présentées dans ce format. Vous ne devez/pouvez pas éditer la bibliographie à la main.

## Bibliographie

- [1] G. B. Ermentrout et L. Edelstein-Keshet, Cellular Automata Approaches to Biological Modeling, *Journal of Theoretical Biology* **160**, 97 (1993).