



**HAL**  
open science

## Les SMA multi-niveaux comme cadre de modélisation et de simulation en épidémiologie

Sébastien Picault, Yu-Lin Huang, Vianney Sicard, François Beaudeau, Pauline Ezanno

► **To cite this version:**

Sébastien Picault, Yu-Lin Huang, Vianney Sicard, François Beaudeau, Pauline Ezanno. Les SMA multi-niveaux comme cadre de modélisation et de simulation en épidémiologie. 25e Journées Francophones sur les Systèmes Multi-Agents (JFSMA 2017), Jul 2017, Caen, France. pp.107-116. hal-01565612

**HAL Id: hal-01565612**

**<https://hal.science/hal-01565612>**

Submitted on 7 Sep 2022

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Les SMA multi-niveaux comme cadre de modélisation et de simulation en épidémiologie

Sébastien Picault<sup>a, b</sup> Yu-Lin Huang<sup>a</sup> Vianney Sicard<sup>a</sup> François Beaudeau<sup>a</sup> Pauline Ezanno<sup>a</sup>

<sup>a</sup>Bioepar, INRA, Oniris, La Chantrerie, 44307 Nantes, France  
prenom.nom@oniris-nantes.fr

<sup>b</sup>Univ. Lille, CNRS, Centrale Lille, UMR 9189 – CRISTAL (équipe SMAC)

## Résumé

*Les modèles épidémiologiques, pour proposer des mesures de maîtrise efficaces, gagnent sans cesse en niveau de détail, et ce à des échelles d'observations multiples, de l'individu aux politiques publiques. Cet élan est freiné par la diversité des paradigmes de modélisation, l'absence de méthodologie logicielle pour l'implémentation de codes toujours plus complexes, et parfois l'existence d'hypothèses implicites. Nous proposons d'utiliser une démarche de modélisation multi-agents multi-niveaux pour intégrer les méthodes existantes au sein d'un cadre générique, forcer une séparation des aspects déclaratifs et procéduraux mais aussi entre domaines d'expertise, et réduire la part de code dévolue aux modélisateurs. Nous illustrons cette démarche par une application sur la maîtrise de la fièvre Q chez les bovins.*

**Mots-clés :** Modélisation multi-niveaux; simulation multi-agents; épidémiologie

## Abstract

*To recommend efficient control measures, epidemiological models incorporate ever-finer details, from individual diversity to public policies, which involve several observation scales. Difficulties in this approach arise from the variety of modelling paradigms, an increased complexity of simulation programs not yet counterbalanced by software engineering methods, and sometimes the existence of implicit assumptions. We propose to use a multi-level agent-based modelling approach to integrate existing methods within a common interface, provide a separation between procedural and declarative concerns but also between expertise fields, and reduce the amount of code left to the designers' responsibility. We illustrate this approach through an application to Q fever control in cattle.*

**Keywords:** Multilevel modelling; Multi-agent-based simulation; Epidemiology

## 1 Introduction

Les travaux que nous présentons s'inscrivent dans le cadre du projet MIMHES<sup>1</sup> qui vise à comprendre les mécanismes de propagation des maladies infectieuses des animaux de production afin de proposer et d'évaluer des stratégies de maîtrise. Dans ce projet, plusieurs maladies enzootiques ont été étudiées à plusieurs échelles d'observation. Divers modèles ont été développés pour chacun des pathogènes cibles, à l'échelle intra-troupeau ou inter-troupeaux, utilisant des modèles à compartiments ou centrés individus, et ce dans des langages de programmation variés (Scilab, C++, Python, R). Ils ont fait la preuve de leur pertinence pour rendre compte des observations de terrain et aidé à concevoir des recommandations en matière de stratégies de maîtrise de ces maladies. Néanmoins, ils manquent de généralité d'un point de vue d'ingénierie logicielle, de sorte que l'exploration de nouvelles hypothèses ou scénarios demande en général d'importants efforts de recodage. Par ailleurs, le choix du paradigme de modélisation (compartiments ou individus) contraint fortement l'architecture du simulateur, alors qu'il serait souvent profitable de pouvoir remplacer des modèles à compartiments par des modèles centrés individus ou vice-versa.

Cette situation est tout à fait représentative des difficultés qui se présentent en modélisation épidémiologique [24], et nous sert donc dans cet article de cas concret pour tester des solutions logicielles nouvelles. Nous pensons notamment que les simulations multi-agents sont un bon candidat pour fournir une interface homogène à plusieurs paradigmes de modélisation. Plus précisément, les concepts et techniques développés dans le cadre des simulations multi-agents multi-niveaux apparaissent parti-

1. « Multi-scale modelling, from animal Intra-Host to Metapopulation, of mechanisms of pathogen spread to Evaluate control Strategies »  
<http://www6.inra.fr/mihmes>

culièrement pertinentes pour les modèles épidémiologiques en raison de la nature intrinsèquement multi-niveaux des processus infectieux, des techniques d'élevage, et des échanges d'animaux.

L'article est organisé de la façon suivante. Dans la section 2, nous rappelons les paradigmes principaux utilisés en modélisation épidémiologique. Les principes de conception et le *framework* proposés en réponse aux problèmes ci-dessus sont présentés dans la section 3, où nous insistons notamment sur la nécessité d'une séparation forte des divers aspects des modèles. Enfin (section 4), nous illustrons notre approche par une application à la maîtrise de la fièvre Q dans les troupeaux bovins laitiers.

## 2 Aperçu des paradigmes de modélisation en épidémiologie

### 2.1 Équations et compartiments

Depuis les travaux fondateurs de Kermack et McKendrick [15], l'approche classique en modélisation épidémiologique consiste à partitionner la population en plusieurs *compartiments* représentant chacun le nombre d'individus partageant un même état. Cette hypothèse forte permet de ne s'intéresser qu'aux flux d'individus passant d'un compartiment à un autre. Un tel modèle est ordinairement représenté au moyen d'un diagramme de flux, par exemple dans le fameux modèle « SIR » (figure 1) constitué de trois états de santé : *sensible* (S), où les individus peuvent s'infecter suite à des contacts, directs ou indirects, avec des individus infectieux ; *infectieux* (I), où les individus sont contagieux ; *rétabli* (R), où les individus ont perdu leur contagiosité et ne peuvent plus être réinfectés. Les trois variables  $S, I, R$  correspondant au nombre d'individus dans chaque compartiment sont contrôlées par un système d'équations différentielles ordinaires reposant directement sur les taux de transition entre compartiments. Elles peuvent être calculées soit de façon continue et déterministe, soit de manière discrète et stochastique, après avoir transformé les taux de transition en probabilités, et en appliquant alors un tirage aléatoire selon une loi multinomiale.

Les modèles compartimentaux, en plus de fournir un éclairage analytique sur le système, sont assez adaptables. Les taux d'entrée et de sortie peuvent représenter des dynamiques démographiques. Les compartiments peuvent être subdivi-



FIGURE 1 – Diagramme de flux pour le modèle classique SIR. Les nœuds sont des compartiments donnant le nombre d'individus dans chaque état de santé. Les arcs sont étiquetés par les taux de transition.

visés en agrégats plus spécifiques, par exemple pour assurer un découpage spatial ou pour modéliser des classes d'âge. Il est également possible de faire coexister plusieurs espèces, comme c'est le cas pour les maladies à vecteurs. De plus, la contamination indirecte par l'environnement peut être explicitement représentée par un ou plusieurs compartiments dédiés [17].

Cependant, ce paradigme est moins adapté pour rendre compte de la diversité rencontrée dans les paramètres biologiques (comme la sensibilité à l'infection), ou pour intégrer des aspects comportementaux, comme les mouvements saisonniers ou les mesures de prévention et de maîtrise. Lorsque plusieurs problématiques se superposent aux états de santé (classes d'âges, conduite de l'élevage, décision de vaccination...), les seules possibilités laissées par cette approche consistent soit à subdiviser encore et encore les compartiments (e.g. [17]), ce qui finit par ressembler beaucoup à un modèle centré individu, soit à complexifier le formalisme [23].

### 2.2 Les modèles centrés individus

En raison de ces limitations, un nombre croissant de travaux font appel aux modèles centrés individus (*Individual-Based Models*, IBM) qui permettent une prise en compte explicite de la diversité des *états individuels* (e.g. [9]), ou même aux modèles multi-agents (*Agent-Based Models*, ABM), offrant dans son principe une diversité des *comportements individuels* et des *interactions* entre individus (e.g. [2, 26]). Les IBM/ABM offrent l'avantage d'une extensibilité quasiment infinie, puisque l'introduction de nouvelles hypothèses se traduit principalement par l'ajout de nouvelles familles d'agents ou de nouveaux comportements. La dynamique du modèle peut être scrutée en détail, et les entités du domaine sont représentées de façon très directe par les entités computationnelles. Il est également plus simple de représenter des processus multifactoriels et de comprendre la nature des mécanismes causaux à l'œuvre dans les systèmes épidémiologiques [18].

En contrepartie, le coût computationnel est nettement plus élevé. Non seulement un IBM est en général plus lent qu'un modèle à compartiments, mais il faut également un plus grand nombre de répétitions, car le niveau de détail se paie de paramètres supplémentaires, qui conduisent donc à une analyse de sensibilité plus poussée. En outre, la facilité à changer des hypothèses mène naturellement à la tentation de multiplier le nombre de scénarios à évaluer afin de comparer des mesures de maîtrise ou des politiques publiques.

D'autre part, tous ces travaux s'intéressent à des cas pratiques et des pathogènes particuliers. Les outils ainsi construits sont pour la plupart *ad hoc* avec dans le meilleur des cas une modélisation objet qui s'abstrait quelque peu du cadre applicatif [27]. Quelques-uns, à l'opposé, s'appuient sur une plateforme de simulation multi-agents, comme GAMA [2] ou NetLogo [26], ce qui constitue un progrès dans la mesure où nombre d'algorithmes (notamment en matière d'ordonancement des agents) ne sont plus à la charge du modélisateur. Toutefois ces plateformes ne sont en rien spécifiques à l'épidémiologie et ne constituent donc pas en elles-mêmes une solution systématique. Force est donc de constater qu'il n'existe à notre connaissance aucune tentative, entre ces deux extrêmes, pour faire des SMA un paradigme de modélisation épidémiologique à part entière, doté d'une méthodologie générale ainsi que d'algorithmes et d'architectures réutilisables.

### 2.3 Vers les simulations multi-niveaux

De nouveaux problèmes se posent lorsque l'on passe à l'échelle régionale ou nationale : les dynamiques spatiales ne peuvent plus être ignorées, et il devient nécessaire de coupler des modèles développés pour des échelles différentes [4].

Une approche classique pour manipuler des modèles épidémiologiques à une échelle régionale repose sur le concept écologique de métapopulation [12]. Une métapopulation est un système de populations locales interconnectées, vivant dans des zones plus ou moins isolées, chacune dotée de sa propre dynamique démographique, voire infectieuse. Les contacts entre populations locales reposent sur les relations de voisinage, les mouvements, ou le transport de pathogènes par des hôtes ou par le vent. Cette méthode réduit le coût computationnel qu'entraînerait la stricte application d'une approche IBM, aux dé-

pens d'une modélisation plus grossière de la dynamique des sous-populations. Elle est très répandue en épidémiologie humaine, où elle permet de traiter des populations et des zones à grande échelle sur la base de structures de contact assez bien décrites (horaires de travail, moyens de transport notamment). De plus, les métapopulations permettent un traitement déterministe comme stochastique. En revanche, comparées à des simulations ABM équivalentes, ces modèles tendent à surestimer la propagation des infections [1, 14].

D'un autre côté, le coût computationnel de simulations multi-agents classiques à cette échelle est considérablement élevé. À moins de faire appel à la fois à des plateformes massivement parallèles, des hypothèses épidémiologiques très simples, et un haut niveau d'optimisation logicielle [22], manipuler des millions ou milliards d'agents est un problème dur, d'autant que de nombreuses répétitions sont nécessaires pour comparer des scénarios de façon significative.

Comme on peut en juger d'après cette diversité des paradigmes de modélisation, il n'existe clairement pas de solution ultime qui serait adaptée à n'importe quelle situation. Au contraire, un cadre de modélisation polyvalent doit être capable de fournir des moyens, méthodologiques aussi bien que logiciels, pour combiner toutes ces méthodes de façon appropriée à chaque contexte de mise en œuvre.

## 3 Une méthode et un framework de modélisation multi-niveaux

Nous partons de l'hypothèse que les modèles doivent être d'emblée *conçus comme multi-niveaux*, plutôt que de construire par exemple un modèle intra-troupeau et tenter de l'étendre à l'échelle régionale, ou au contraire de chercher à partir d'un modèle de métapopulation en rajoutant à chaque troupeau des caractéristiques plus fines. Ce parti-pris, loin d'être utopique, peut être réalisé en s'appuyant sur une forte modularité en particulier en ce qui concerne 1) la *structure* des modèles, pour laquelle une approche multi-agents multi-niveaux semble bien adaptée, et 2) une décomposition explicite de tous les processus à l'œuvre dans le système (dynamique infectieuse, gestion du troupeau, échanges commerciaux...). Dans cette section, nous présentons les principes de conception et l'architecture logicielle que nous proposons en réponse à ces problématiques.

### 3.1 Modularité structurelle : modélisation multi-niveaux

En épidémiologie animale, la possibilité de réifier des entités intermédiaires entre l'animal et la population (groupes d'âge, troupeaux...), ainsi que des zones spatiales correspondantes (enclos, fermes, pâturages...) constitue un point clef pour évaluer les mécanismes fins de mesures de maîtrise. Or, le nombre de travaux de recherche sur les simulations multi-agents multi-niveaux est en forte croissance depuis quelques années. Ces méthodes consistent à faire appel à des agents pour réifier des niveaux d'organisation, d'observation ou d'échelle au sein même de la simulation en cours. Néanmoins la plupart de ces contributions sont conçues pour des domaines d'applications spécifiques et s'appuient sur des méthodes *ad hoc*.

Aussi, nous avons choisi de nous appuyer sur un des rares méta-modèles multi-niveaux génériques existants (avec principalement [8, 10, 20]), à savoir PADAWAN [25], qui offre des caractéristiques adaptées aux besoins en épidémiologie computationnelle. En particulier, ce méta-modèle impose une *forte séparation entre les aspects déclaratifs et procéduraux*, à travers notamment l'indépendance entre les comportements (spécifiés comme règles d'interaction génériques) et les agents amenés à les réaliser. Les agents et les environnements peuvent être associés via deux relations : la *situation* (qui permet aux agents d'interagir au sein d'un ou plusieurs environnements) et l'*encapsulation* (un agent peut « contenir » un environnement pour héberger d'autres agents). L'utilisation conjointe de ces deux relations permet de définir la notion d'*hébergement* (*a* héberge *b* si *b* est situé dans un environnement encapsulé par *a*).

Ainsi, un SMA multi-niveaux est construit comme *combinaison d'une structure* (l'architecture et l'organisation des agents et des environnements emboîtés) *et d'une fonction* (une description explicite et intelligible des processus qui se déroulent dans le système). Comme les agents peuvent représenter n'importe quelle sorte d'entité, ils fournissent une interface homogène mais polymorphe pour intégrer de multiples paradigmes de modélisation.

### 3.2 Modularité fonctionnelle : gestion des connaissances

Nous estimons par ailleurs que les connaissances du domaine introduites dans un modèle

(paramètres, hypothèses, processus, données...) doivent être segmentées en fonction de la diversité des points de vue, des objectifs et des expertises. Nous nous situons en cela dans le même esprit que les travaux de [6, 7] dans le cadre de modèles à compartiments, qui distinguent par exemple les problématiques liées au processus infectieux, à la gestion des espèces et à la distribution spatiale. Nous souhaitons aller plus loin dans cette voie en permettant l'introduction de mécanismes *a priori* quelconques, y compris extérieurs à l'épidémiologie *stricto sensu*, comme la conduite des élevages, les aspects économiques de la prise de décision par l'éleveur, le vétérinaire ou les pouvoirs publics, les effets physiologiques des traitements, etc. La première tâche dans la conception d'un modèle consiste donc à identifier l'ensemble de ces domaines et de leurs processus.

Pour permettre à des experts épidémiologistes d'être pleinement acteurs de la modélisation, il est en outre nécessaire que le cadre conceptuel qui leur est proposé soit *aussi peu intrusif que possible* par rapport aux outils utilisés habituellement, de façon à rendre plus précises les méthodes existantes sans demander de gros efforts d'adaptation. Nous proposons donc les principes suivants :

1. *Améliorer les formalismes existants pour réduire leurs ambiguïtés, sans les rendre abscons.* Par exemple, le protocole ODD proposé par [13] (principalement pour l'écologie) est un premier pas vers l'explicitation des connaissances des experts, mais c'est avant tout un cadre textuel qui reste ambigu [3]. Au contraire, l'utilisation de formalismes très riches issus soit de la physique multi-échelles, comme les *Bond Graphs* [11], soit de la biologie moléculaire, comme SBGN [16], serait inapproprié au regard des préoccupations épidémiologiques car d'une complexité disproportionnée.
2. *Fournir des guides méthodologiques pour apporter à des problèmes typiques une réponse pertinente et standardisée,* comme le font les *Design Patterns* en Génie Logiciel.
3. *Automatiser autant que possible les tâches centrales du processus de simulation,* de façon à n'imposer aux utilisateurs finaux le développement que de portions de code spécifiques et de petite taille.

Afin de maintenir une proximité avec les formalismes existants, nous proposons de transformer le diagramme de flux utilisé dans les

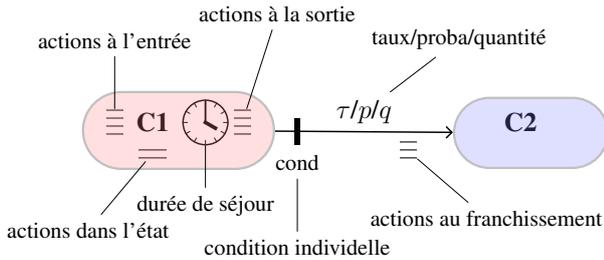


FIGURE 2 – Extension du diagramme de flux sous la forme d’une machine à états.

modèles à compartiments en une véritable machine à états finis. Pour ce faire, nous considérons que les nœuds et les arcs peuvent être dotés d’informations complémentaires (fig. 2). Les états peuvent recevoir des caractéristiques facultatives : 1) une *distribution des durées de séjour*, qui spécifie combien de temps un individu est susceptible de rester dans l’état courant, et qui peut être particulièrement utile pour décrire des effets démographiques ; 2) des *actions* exécutées à l’entrée, durant le séjour, ou à la sortie de l’état, par exemple pour gérer l’excrétion dans les états infectieux. En plus de leur étiquette qui représente soit un taux, soit une probabilité, soit un nombre absolu d’individus, on peut ajouter aux transitions : 1) *des conditions de franchissement*, pour déterminer quels agents sont autorisés à migrer de l’état source vers l’état de destination, et 2) des *actions* exécutées par les individus qui franchissent l’arc, i.e. après avoir quitté l’état source et avant d’entrer dans l’état de destination.

Ces ajouts correspondent en fait à des informations « métier » qui, étant absentes des modèles spécifiés par un diagramme de flux classique, sont d’ordinaire codées directement lors de l’implémentation. Déplacés vers l’amont dans la phase conception, ces éléments contribuent à une vision plus précise du modèle dans son ensemble, et peuvent être inclus dans un processus de génération de code. De plus, ce diagramme de machine à états peut être utilisé indifféremment pour décrire ce qui se passe dans un modèle à compartiments (déterministe ou stochastique) ou centré individus.

Toutes les informations concernant les processus d’un modèle, les machines à état correspondants avec leurs états, transitions, conditions, actions, durées et les paramètres de taux, probabilités ou quantités sont spécifiées au moyen d’un fichier de configuration au format YAML (choisi pour sa grande lisibilité), qui est traité pour générer l’architecture de simulation mais

peut également servir à produire une documentation technique, des figures, inclure des commentaires sur les sources utilisées et les hypothèses sous-jacentes, etc.

### 3.3 Architecture du framework EMuLSion

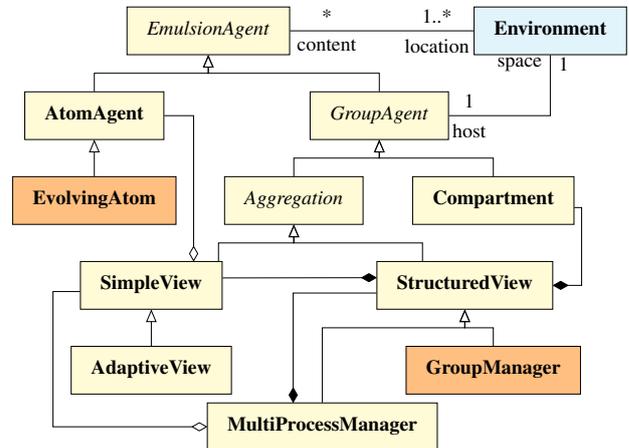


FIGURE 3 – Diagramme de classes de la hiérarchie des agents multi-niveaux dans le framework EMuLSion. Le comportement des agents de classes à fond orangé est contrôlé par une machine à états.

Ces principes de conception ont été implémentés et expérimentés à travers un framework réalisé en Python, appelé « EMuLSion<sup>2</sup> ». Dans ce framework, un modèle est composé de processus (séquences d’actions sur les agents), certains d’entre eux étant dirigés par des machines à états. Ils s’appuient sur un ensemble de paramètres clairement décrits comme les taux, probabilités, quantités, distributions, etc. donnés en valeurs, en intervalles, ou exprimés en fonction d’autres paramètres. Ces informations sont analysées au moyen d’une bibliothèque de calcul symbolique (SymPy) et la cohérence du modèle est vérifiée automatiquement.

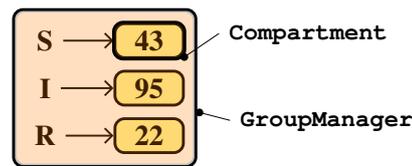
Les classes des agents utilisés dans EMuLSion sont présentées sur la figure 3. Tous les agents multi-niveaux sont situés dans au moins un environnement. Ils se répartissent en deux grandes catégories : les *atomes* et les *groupes*. La classe AtomAgent représente des « individus » (e.g. les animaux). Sa sous-classe EvolvingAtom correspond aux individus dont le comportement est régi par des machines à états. De leur côté, les groupes d’agents peuvent encapsuler un environnement local où d’autres agents peuvent se situer. Les groupes sont composés eux-mêmes de deux familles : les *compartiments* et les

2. pour *Epidemiological Multi-Level Simulation framework*

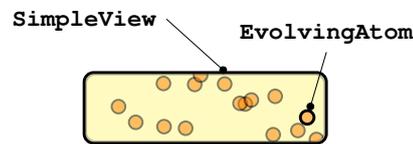
*agrégats*. La classe `Compartment` est utilisée lorsque les individus sont suffisamment homogènes pour être décrits par une simple quantité. Une `Aggregation` fournit au contraire une *vue* sur des individus ou d'autres groupes, i.e. une représentation où les agents sont rassemblés en fonction de variables identifiées au préalable comme pertinentes selon un point de vue donné, comme l'état de santé, le groupe d'âge, etc.

Parmi ces agrégats, on distingue d'abord les agents `SimpleView` qui hébergent des individus (atomes) et gèrent l'ordonnancement de leurs comportements. Une `AdaptiveView` détecte en outre les individus dont les variables clefs ont une valeur différente des autres, et demandent à leur hôte de les placer dans un groupe adéquat. Ensuite, un agent `StructuredView` a pour fonction d'associer des agents `SimpleView` ou `Compartment` aux valeurs possibles de variables clefs. Par exemple, dans un modèle SIR, la variable `health_state` peut prendre trois valeurs (S, I, R), et se trouve donc associée à trois agents `Compartment` ou à trois agents `SimpleView`. Le `GroupManager` fait de même, mais dispose en outre d'une machine à états pour déterminer quels individus doivent changer de valeur pour ces variables clefs. Enfin, un agent `MultiProcessManager` peut gérer plusieurs processus dans une même simulation. Il s'appuie au minimum sur un agent `SimpleView` pour superviser tous les individus, et sur un agent (en fait, souvent plusieurs) `StructuredView` ou `GroupManager` pour gérer les différents processus du modèle.

On notera incidemment que cette architecture s'appuie délibérément sur les quatre *Design Patterns* pour les SMA multi-niveaux définis dans [19]. Le `Compartment` correspond à une agrégation destructive d'agents micro en un agent macro (pattern ZOOM). Les agents `SimpleView` et `StructuredView` ne font que représenter au niveau macro l'état des agents micro qu'ils hébergent (pattern VUE). À l'opposé, le `GroupManager`, en charge d'une machine à états propre à une problématique, dépouille les agents micro qu'il héberge de leur autonomie comportementale au regard de ce processus précis et se charge de les piloter (pattern Marionnettiste). Le dernier pattern (Cohabitation) correspond à une situation où un `MultiProcessManager` en héberge d'autres, par exemple une métapopulation et ses sous-populations : dans ce cas certains processus macroscopiques (échanges commerciaux) peuvent agir comme un forçage sur les agents



(a) Modèle à compartiments



(b) Modèle centré individus

FIGURE 4 – (a) Structure d'un modèle à compartiments. Le `GroupManager` utilise une machine à états pour déterminer les flux entre compartiments. (b) Structure d'une simulation IBM. Les `EvolvingAtoms` possèdent leur propre machine à états et sont hébergés par un agent `SimpleView`.

microscopiques, sans pour autant priver ces derniers de leur autonomie.

### 3.4 Modes d'utilisation

Cette architecture qui repose sur les capacités de composition des agents permet en pratique de représenter les paradigmes de modélisation épidémiologique classiques. Les *modèles à compartiments* sont construits en utilisant des agents `Compartment` hébergés par un `GroupManager`, lui-même doté d'une machine à états qui détermine les états de santé et leurs transitions (fig. 4a). Les agents `Compartment` peuvent adopter un comportement déterministe ou stochastique à la demande. Les *modèles centrés individus* sont composés d'agents `EvolvingAtom`, chacun doté d'une (ou plusieurs) machine(s) à états, et hébergés par un agent `SimpleView` qui ordonnance leur activité (fig. 4b). Enfin, les *métapopulations* peuvent être réalisées au moyen d'un agent `StructuredView` ou `MultiProcessManager` contenant plusieurs autres agents construits selon l'une ou l'autre des approches précédentes, avec en plus des processus dédiés à la description des structures de contact.

Mais d'autres solutions peuvent être élaborées, en particulier en regroupant les individus en fonction de leur état, pour chacun des aspects modélisés. L'intérêt principal de tels regroupements est de fournir un accès direct à des individus jugés similaires selon un certain point de vue. En particulier, pour déter-

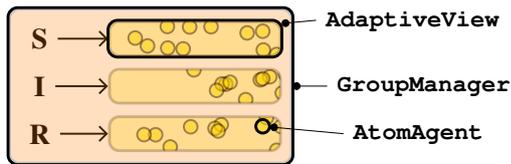


FIGURE 5 – Structure du regroupement d’individus au sein de vues adaptatives. Les atomes sont répartis en fonction de leurs états communs. Le GroupManager est doté d’une machine à états pour piloter les changements des atomes.

miner lesquels doivent changer d’état, il suffit dans une approche stochastique d’un seul tirage selon une loi multinomiale par groupe, au lieu d’une épreuve de Bernoulli par individu, d’où une efficacité accrue. Pour réaliser ces regroupements dynamiques, la première étape consiste à construire un agent GroupManager équipé d’une machine à états pour un aspect du modèle, tel que le processus infectieux qui impacte la variable `health_state` (fig. 5). Chaque valeur possible de `health_state` est associée à un agent AdaptiveView, contenant des AtomAgents. Les atomes changent d’état de santé selon la machine à états du GroupManager, mais si un processus extérieur affecte leur variable `health_state`, par exemple un traitement qui transforme des « I » en « R », l’agent AdaptiveView en charge de l’état « I » détecte ces modifications et demande au GroupManager de déplacer ces atomes modifiés à la bonne place (chez l’agent AdaptiveView en charge de l’état « R »).

L’agent MultiProcessManager se charge de coordonner plusieurs structures analogues (cf. section suivante, fig. 7). Il reçoit la liste des processus affectant les individus ou les groupes, ainsi que les variables associées à chaque regroupement. Si un processus est lié à une machine à états, il fait automatiquement appel à un GroupManager (structuré comme indiqué ci-dessus), sinon il fait de même avec un simple StructuredView. Chaque individu est donc accessible à partir d’une structure globale (SimpleView) et de chacun des regroupements.

Ce framework a été testé en détail sur plusieurs variations de modèles théoriques du type SIR, pour vérifier que tous les cas d’usages décrits ci-dessus permettaient de reproduire des résultats équivalents. Dans la section qui suit, nous montrons une application à un cas réel, la fièvre Q.

## 4 Application à la fièvre Q chez les bovins

Afin de donner une illustration concrète de notre approche, nous allons montrer comment elle a été appliquée à une des maladies étudiées dans le projet MIHMES : la fièvre Q, une zoonose qui touche principalement les ruminants. Deux modèles centrés individus ont été développés auparavant pour cette maladie dans le cadre des bovins et servent de référence : l’un pour la propagation intra-troupeau [9] (en R), l’autre pour la propagation inter-troupeaux [21] (en Python).

### 4.1 Le modèle fièvre Q

Notre implémentation du modèle de la fièvre Q dans les troupeaux bovins laitiers s’appuie sur les travaux antérieurs [9] avec 6 états de santé : sensible (S), infectieux sans réponse immunitaire (I-), infectieux avec réponse immunitaire (I+) voire excrétion bactérienne dans le lait (I+m), et porteur avec (C+) ou sans (C-) anticorps. La contamination se fait par dépôt de bactéries dans l’environnement. Par ailleurs, seules les femelles adultes sont prises en compte. Elles sont soumises à une conduite d’élevage (ou « cycle de vie ») adaptée à la production laitière, au cours de laquelle les vaches sont inséminées, deviennent gestantes (état P), vêlent (PC) ou avortent (A) selon leur état de santé, et attendent la gestation suivante (BP). Le vêlage et l’avortement entraînent un haut niveau d’excrétion bactérienne. Ces processus sont représentés sur la figure 6.

La simulation s’effectue en temps discret. À chaque pas de temps, les processus impliqués au niveau du troupeau sont les suivants : 1) mise à jour des bactéries présentes dans l’environnement (décroissance exponentielle); 2) réforme des vaches (retrait du troupeau) en fonction de leur parité (nombre de vêlages antérieurs); 3) remplacement pour introduire de nouveaux animaux dans le troupeau; 4) infection pilotée par la machine à états affectant l’état de santé (fig. 6a); 5) conduite de l’élevage pilotée par celle affectant le cycle de vie (fig. 6b); 6) mise à jour du regroupement des animaux par parité.

### 4.2 Implémentation, tests et validation

L’implémentation de ce modèle pourrait suivre une approche IBM classique, mais pour bénéficier du regroupement adaptatif des individus,

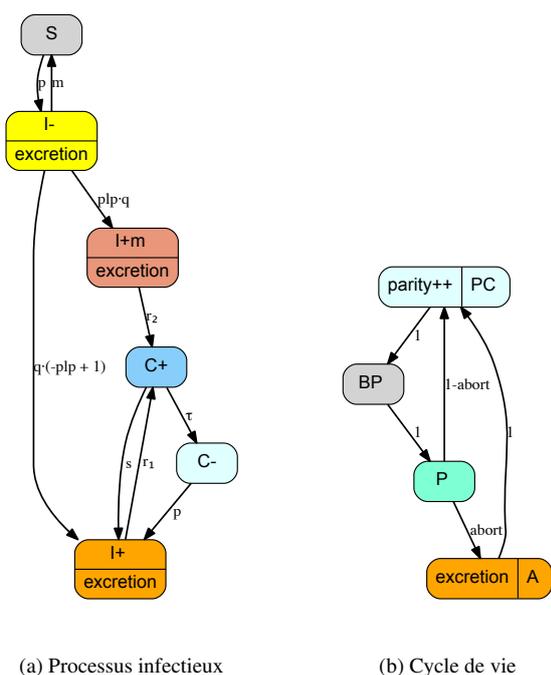


FIGURE 6 – Machines à états pour le processus infectieux et le cycle de vie des vaches adultes. Les actions à l'entrée/au cours des états sont indiquées. Les valeurs des arcs sont des probabilités.

l'architecture la plus appropriée consiste à définir d'une part une classe pour les individus (QfeverCow, dérivée de AtomAgent), et une pour le troupeau (QfeverHerd, sous-classe de MultiProcessManager).

Les processus impliqués dans la fièvre Q conduisent à regrouper les individus selon trois critères : la parité (pour la réforme), le cycle de vie et l'état de santé (ces deux dernières variables étant contrôlées par une machine à états). La quantité de bactéries excrétées par les animaux infectieux est en outre plus élevée dans les semaines qui suivent le vêlage, aussi est-il plus efficace d'utiliser une variable booléenne supplémentaire (recent\_calving) pour contrôler le regroupement lié au processus infectieux. Cela pourrait multiplier les agents AdaptiveView en raison de la combinatoire des valeurs possibles de ces variables, mais en pratique le GroupManager n'en crée que lorsque c'est nécessaire pour héberger des atomes. L'architecture du SMA qui résulte de ces contraintes est présentée sur la figure 7.

La figure 8 montre que ce modèle implémenté dans EMuLSion reproduit les résultats du modèle de référence [9], c'est-à-dire une coïncidence des moyennes et des percentiles des prin-

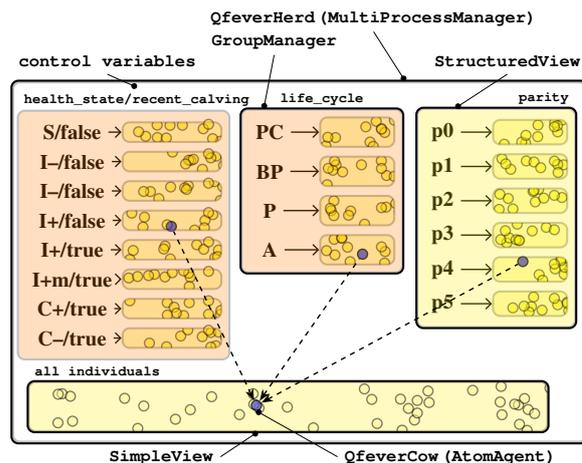


FIGURE 7 – Structure du modèle intra-troupeau de la fièvre Q. Les individus sont rassemblés en fonction des valeurs des variables clefs de chaque aspect du modèle. On accède aux individus (e.g. celui représenté en bleu, dans l'état de santé  $I^+$ , le cycle de vie A et de parité 4) soit par aspect, soit par l'agent SimpleView hébergeant tous les atomes.

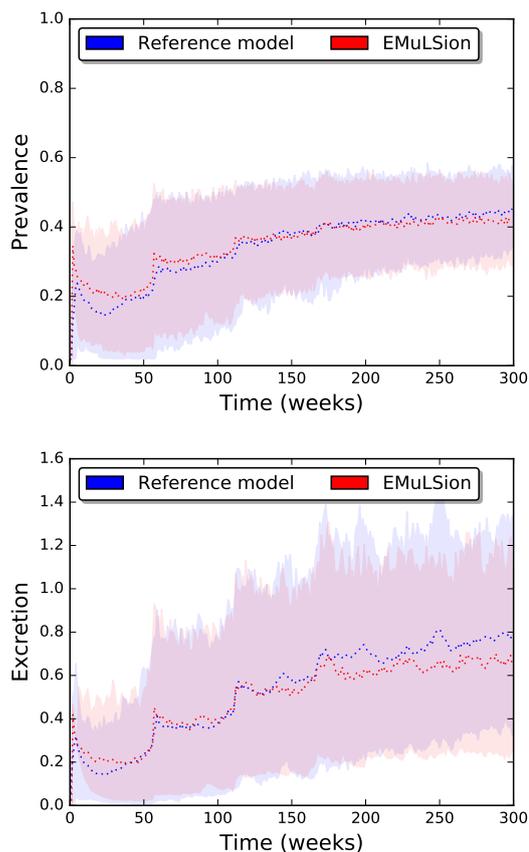


FIGURE 8 – Évolution dans le temps de la prévalence et de l'excrétion dans l'environnement, en moyenne et percentiles sur 200 simulations, pour le modèle de référence [9] et l'implémentation dans EMuLSion pour la fièvre Q.

cipales variables de sortie pertinentes d'un point de vue épidémiologique (ici, l'excrétion dans l'environnement et la prévalence, i.e. la proportion d'animaux infectieux dans le troupeau), calculées pour un nombre suffisamment important de répétitions (ici 200).

En outre, la séparation entre les aspects procéduraux et déclaratifs, mais aussi entre les divers processus du modèle, a permis de faciliter grandement la formulation et la vérification d'hypothèses alternatives, en particulier l'étude des simplifications possibles du modèle intra-troupeau pour ne garder que les mécanismes essentiels à l'explication causale de la propagation de la maladie, avant de passer à la contamination inter-troupeaux. Ce travail exploratoire fera l'objet d'une publication prochaine dans le domaine épidémiologique.

## 5 Conclusion et perspectives

Nous avons présenté une approche générique pour la conception et l'implémentation de modèles épidémiologiques, basée sur une structure d'hébergement d'agents répartis dans plusieurs niveaux d'organisation. Notre méthode offre une façon homogène d'intégrer facilement des paradigmes de modélisation classiques comme les modèles à compartiments ou centrés individus. Il s'appuie pour cela sur une séparation forte entre les aspects procéduraux et déclaratifs des modèles, ainsi qu'entre les divers processus et domaines d'expertise nécessaires. Ainsi, les hypothèses sous-jacentes sont toutes explicitées sous une forme intelligible et révisable, et peuvent être traitées par un moteur de simulation générique et d'une fiabilité accrue.

Ce framework a été d'ores et déjà mis en œuvre pour refondre le modèle intra-troupeau de la fièvre Q et tester quelles hypothèses peuvent être simplifiées, de façon à construire un modèle inter-troupeaux efficace et pertinent. Les modèles de référence, difficilement modifiables, comptent environ 1 000 lignes de code R (intra-troupeau) et 2 500 de code Python (inter-troupeaux). La version actuelle du framework compte environ 3 200 lignes de code Python, ce qui permet de réduire les développements spécifiques à la fièvre Q à seulement 250 lignes. Le fichier YAML qui décrit les machines à états, les processus et les paramètres, est composé d'environ 300 entrées clef-valeur (en incluant les commentaires, descriptions et sources bibliographiques pour chacun d'eux).

Les travaux en cours portent sur l'intégration de ce modèle dans une simulation multi-troupeaux, au moyen d'un agent `MultiProcessManager` pour représenter la métapopulation de troupeaux. La propagation de la fièvre Q entre troupeaux est causée par la diffusion des bactéries sous forme d'aérosols ainsi que par l'introduction d'animaux infectés au cours des échanges commerciaux. Cela n'impacte donc que deux processus du modèle intra-troupeau : la mise à jour des quantités de bactéries dans l'environnement (transport par le vent sur la base de données météorologiques), et le remplacement des animaux (à partir des données de mouvements par ventes et achats). Le volume de code à écrire est donc lui aussi très faible.

Les autres modèles développés au cours du projet MIHMES vont faire l'objet de travaux similaires pour assurer leur durabilité et faciliter le développement d'extensions, la révision d'hypothèses, et l'étude de nouveaux scénarios de maîtrise. Plus généralement, ce travail est un premier pas vers l'élaboration collective d'un cadre commun de modélisation qui permette une comparaison précise et factuelle des modèles épidémiologiques. La possibilité de rendre toutes les hypothèses explicites et de décrire finement tous les processus et interactions impliquées dans un système, ainsi que la confiance dans un moteur de simulation à la fois générique et flexible, sont une exigence cruciale pour la reproductibilité scientifique. Dans ce but, EMuLSion sera à terme distribué en *open source*. Par ailleurs, la capacité à utiliser indifféremment et de façon transparente l'un ou l'autre des multiples paradigmes existants ouvre la voie à l'exploration de techniques visant à basculer automatiquement de l'un à l'autre, comme suggéré dans [5].

Dans une perspective plus large, nous pensons aussi que la réduction des durées de conception, d'implémentation et de validation de nouveaux modèles contribue de façon essentielle à une réactivité accrue pour répondre aux urgences épidémiologiques (grippe humaine ou aviaire, maladies transmises par les moustiques), enjeu majeur face à la multiplication de situations inédites consécutives aux changements globaux en cours et à venir.

## Remerciements

Ces travaux sont réalisés dans le cadre du projet MIHMES, co-financé par l'ANR (investis-

sements d’avenir ANR-10-BINF-07) et par le Fonds Européen de Développement Régional (FEDER) des Pays-de-la-Loire. Ils bénéficient également d’un financement du département Santé Animale de l’INRA.

## Références

- [1] M. Ajelli, B. Gonçalves, D. Balcan, V. Colizza, H. Hu, J.J. Ramasco, S. Merler, and A. Vespignani. Comparing large-scale computational approaches to epidemic modeling : Agent-based versus structured metapopulation models. *BMC Infectious Diseases*, 10(1), 2010.
- [2] E. Amouroux, S. Desvaux, and A. Drogoul. Towards virtual epidemiology : An agent-based approach to the modeling of H5N1 propagation and persistence in north-vietnam. In *11th Pacific Rim Int. Conf. on Multi-Agents (PRIMA)*, volume 5357 of LNCS, pages 26–33. Springer, 2008.
- [3] E. Amouroux, B. Gaudou, S. Desvaux, and A. Drogoul. O.D.D. : A promising but incomplete formalism for individual-based model specification. In *IEEE RIVF Int. Conf. on Computing and Communication Technologies*. IEEE, 2010.
- [4] G. Beaunée, E. Vergu, and P. Ezanno. Modelling of paratuberculosis spread between dairy cattle farms at a regional scale. *Veterinary Research*, 46(1), 2015.
- [5] G.V. Bobashev, D.M. Goedecke, F. Yu, and J.M. Epstein. A hybrid epidemic model : Combining the advantages of agent-based and equation-based approaches. *Winter Simulation Conference*, 2007.
- [6] T.M.A. Bui, S. Stinckwich, M. Ziane, B. Roche, and T.V. Ho. KENDRICK : A domain specific language and platform for mathematical epidemiological modelling. In *IEEE RIVF Int. Conf. on Computing and Communication Technologies*, pages 132–137. IEEE, 2015.
- [7] T.M.A. Bui, M. Ziane, S. Stinckwich, T.V. Ho, B. Roche, and N. Papoulias. Separation of concerns in epidemiological modelling. In *15th Int. Conf. on Modularity*, pages 196–200. ACM, 2016.
- [8] B. Camus, C. Bourjot, and V. Chevrier. Multi-level modeling as a society of interacting models. In *Agent-Directed Simulation Symposium (in Spring-Sim)*. SCS/ACM, 2013.
- [9] A. Courcoul, H. Monod, M. Nielen, D. Klinkenberg, L. Hogerwerf, F. Beaudeau, and E. Vergu. Modeling the effect of heterogeneity of shedding on the within herd *Coxiella burnetii* spread and identification of key parameters by sensitivity analysis. *J. Theor. Biol.*, 284(1) :130–141, 2011.
- [10] A. Drogoul, E. Amouroux, P. Caillou, B. Gaudou, A. Grignard, N. Marilleau, P. Taillandier, M. Vavasseur, D.-A. Vo, and J.-D. Zucker. GAMA : multi-level and complex environment for agent-based models and simulations. In *AAMAS*, pages 1361–1362, 2013.
- [11] V. Díaz-Zuccarini and C. Pichardo-Almarza. On the formalization of multi-scale and multi-science processes for integrative biology. *Interface Focus*, 1(3) :426–437, 2011.
- [12] B. Grenfell and J. Harwood. (meta)population dynamics of infectious diseases. *Trends in Ecology & Evolution*, 12(10) :395–399, 1997.
- [13] V. Grimm and et al. A standard protocol for describing individual-based and agent-based models. *Ecological Modelling*, 198(1–2) :115–126, 2006.
- [14] M.J. Keeling, L. Danon, M.C. Vernon, and T.A. House. Individual identity and movement networks for disease metapopulations. *PNAS*, 107(19) :8866–8870, 2010.
- [15] W.O. Kermack and A.G. McKendrick. A contribution to the mathematical theory of epidemics. *Proc. R. Soc.*, (A115) :700–721, 1927.
- [16] N. Le Novère and et al. The systems biology graphical notation. *Nat. Biotech.*, 27(8) :735–741, 2009.
- [17] C. Marcé, P. Ezanno, H. Seegers, D. Pfeiffer, and C. Fourichon. Predicting fadeout versus persistence of paratuberculosis in a dairy cattle herd for management and control purposes : a modelling study. *Veterinary Research*, 42(1) :36, 2011.
- [18] B.D.L. Marshall and S. Galea. Formalizing the role of agent-based modeling in causal inference and epidemiology. *Am. J. Epidemiology*, 181(2) :92–99, 2014.
- [19] P. Mathieu, G. Morvan, and S. Picault. Simulations multi-agents multi-niveaux : quatre patterns de conception. In *24e Journées Francophones sur les Systèmes Multi-Agents (JFSMA)*, pages 117–126. Cépaduès, 2016. (présentation courte).
- [20] G. Morvan, A. Veremme, and D. Dupont. IRM4MLS : The influence reaction model for multi-level simulation. In *Multi-Agent-Based Simulation XI*, volume 6532 of LNCS, pages 16–27. Springer Nature, 2011.
- [21] P. Pandit, T. Hoch, P. Ezanno, F. Beaudeau, and E. Vergu. Spread of *Coxiella burnetii* between dairy cattle herds in an enzootic region : modelling contributions of airborne transmission and trade. *Veterinary Research*, 47(1), 2016.
- [22] J. Parker and J.M. Epstein. A distributed platform for global-scale agent-based models of disease transmission. *ACM Trans. Model. Comput. Simul.*, pages 2 :1–2 :25, 2011.
- [23] N. Perra, D. Balcan, B. Gonçalves, and A. Vespignani. Towards a characterization of behavior-disease models. *PLoS ONE*, 6(8) :e23084, 2011.
- [24] S. Picault, Y.-L. Huang, V. Sicard, and P. Ezanno. Enhancing sustainability of complex epidemiological models through a generic multilevel agent-based approach. In *26th Int. Joint Conf. on Artificial Intelligence (IJCAI)*. AAAI, 2017. (in press).
- [25] S. Picault and P. Mathieu. An interaction-oriented model for multi-scale simulation. In *22nd Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 332–337, 2011.
- [26] J. Robins, S. Bogen, A. Francis, A. Westhoek, A. Kanarek, S. Lenhart, and S. Eda. Agent-based model for John’s disease dynamics in a dairy herd. *Veterinary Research*, 46(1), 2015.
- [27] B. Roche, J.-F. Guégan, and F. Bousquet. Multi-agent systems in epidemiology : a first step for computational biology in the study of vector-borne disease transmission. *BMC Bioinformatics*, 9, 2008.