



HAL
open science

Modèle multi-agent d'auto-organisation pour le butinage au sein d'une colonie d'abeilles

Jérémy Rivière, Thomas Alves, Cédric Alaux, Yves Le Conte, Yves Layec,
André Lozac'h, Frank Singhoff, Vincent Rodin

► To cite this version:

Jérémy Rivière, Thomas Alves, Cédric Alaux, Yves Le Conte, Yves Layec, et al.. Modèle multi-agent d'auto-organisation pour le butinage au sein d'une colonie d'abeilles. *Revue Ouverte d'Intelligence Artificielle*, 2022, 3 (5-6), pp.423 - 450. 10.5802/roia.38 . hal-03868878

HAL Id: hal-03868878

<https://hal.science/hal-03868878>

Submitted on 24 Nov 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.



JÉRÉMY RIVIÈRE, THOMAS ALVES, CÉDRIC ALAUX, YVES LE CONTE,
YVES LAYEC, ANDRÉ LOZAC'H, FRANK SINGHOFF, VINCENT RODIN

Modèle multi-agent d'auto-organisation pour le butinage au sein d'une colonie
d'abeilles

Volume 3, n° 5-6 (2022), p. 423-450.

<https://doi.org/10.5802/roia.38>

© Les auteurs, 2022.



Cet article est diffusé sous la licence
CREATIVE COMMONS ATTRIBUTION 4.0 INTERNATIONAL LICENSE.
<http://creativecommons.org/licenses/by/4.0/>



*La Revue Ouverte d'Intelligence Artificielle est membre du
Centre Mersenne pour l'édition scientifique ouverte*
www.centre-mersenne.org
e-ISSN : pending

Modèle multi-agent d'auto-organisation pour le butinage au sein d'une colonie d'abeilles

Jérémy Rivière^a, Thomas Alves^a, Cédric Alaux^b, Yves Le Conte^b,
Yves Layec^c, André Lozac'h^c, Frank Singhoff^a, Vincent Rodin^a

^a Univ Brest, Lab-STICC, CNRS, UMR 6285, F-29200 Brest, France

E-mail : jeremy.riviere@univ-brest.fr, thomas.alves@univ-brest.fr

^b INRAE, UR 406 Abeilles et Environnement, Avignon, France

E-mail : cedric.alaux@inra.fr, yves.leconte@paca.inra.fr

^c Groupement de Défense Sanitaire Apicole du Finistère (GDSA29), France

E-mail : yves.layec@orange.fr, andre.lozach3@free.fr, frank.singhoff@univ-brest.fr,
vincent.rodin@univ-brest.fr.

RÉSUMÉ. — Les Systèmes Multi-Agents (SMA) ont montré depuis plusieurs années leur adéquation à modéliser et simuler les systèmes complexes. Nous suivons cette approche pour modéliser une colonie d'abeilles située dans une ruche Dadant, où plusieurs dizaines de milliers d'individus interagissent, dans le but d'évaluer l'impact d'actions locales au niveau des abeilles (*e.g.* pratiques apicoles) sur la colonie. Nous nous concentrons ici sur l'activité de butinage, en nous intéressant plus particulièrement au phénomène d'auto-organisation qui conduit les butineuses à sélectionner les meilleures sources de nourriture disponibles. Les interactions des butineuses avec l'environnement extérieur de la ruche, qui diffère de l'intérieur en termes de granularité des actions et d'échelle, sont simulées grâce à un module paramétrable et compatible agent, en fonction de la météo et des sources de nourriture environnantes. Les résultats de deux expérimentations du modèle, l'une sur une année complète, et l'autre sur une journée, montrent que le phénomène d'auto-organisation des butineuses résulte du comportement des butineuses et des mécanismes de recrutement implantés, et offrent une première validation de notre modèle.

MOTS-CLÉS. — Simulation multi-agent, Auto-organisation, Systèmes Complexes, Environnement.

1. INTRODUCTION

Les systèmes complexes peuvent être définis comme « des systèmes dans lesquels de grands réseaux de composants, sans contrôle central, et suivant de simples règles, donnent naissance à des comportements collectifs complexes » [19]. Parce qu'ils se concentrent sur l'individu pour obtenir par émergence le comportement global,

les Systèmes Multi-Agents (SMA) sont particulièrement adaptés pour modéliser et simuler de tels systèmes.

Une colonie d’abeilles, dans laquelle plusieurs dizaines de milliers d’individus interagissent et vivent ensemble, est un bon exemple de système complexe. En effet, une des caractéristiques notables d’*Apis mellifera Linnaeus*, utilisée en apiculture, et plus largement des abeilles du genre *Apis*, est qu’elles vivent en colonie suivant un comportement eusocial⁽¹⁾ [35]. Cette organisation sociale dépend des multiples interactions (e.g. phéromones [31], nutrition, danse) entre des individus spécialisés (butineuses, nourrices, etc.) d’âges et de castes différents. Des phénomènes complexes émergent de ces interactions au niveau du système (la colonie), permettant son auto-régulation et son auto-adaptation, comme par exemple :

- Le changement de fonction des individus suivant les besoins de la colonie [25];
- La thermorégulation du nid [29];
- La sélection des meilleures sources de nectar disponibles [11, 27], à laquelle cet article s’intéresse.

Notre projet⁽²⁾ est de modéliser ces phénomènes pour simuler une colonie d’abeilles avec deux objectifs. Premièrement, nous souhaitons offrir à un utilisateur des moyens d’interagir avec une ruche virtuelle, directement au niveau des abeilles, pour deux usages différents : dans un objectif pédagogique (c’est-à-dire pour apprendre les bonnes pratiques en apiculture), et dans un objectif scientifique afin d’évaluer les phénomènes d’ordre écologiques (parasites, virus) et anthropiques. Deuxièmement, nous souhaitons découvrir ou mieux comprendre les mécanismes menant à l’émergence de ces phénomènes afin de contribuer à résoudre des problèmes similaires dans d’autres contextes. Cette démarche, déjà appliquée à certaines espèces animales vivant en sociétés, a été menée avec succès à plusieurs occasions [8, 12]. Dans le cas des abeilles, leur capacité à détecter les sources de nourriture les plus profitables pour rapidement les exploiter avant qu’elles ne disparaissent est très efficace, et pourrait inspirer de nouvelles méthodes d’organisation pour la résolution de problèmes d’allocation dynamique de ressources (critiques).

Certains de ces phénomènes peuvent être modélisés au niveau de la colonie sans considérer les abeilles comme des individus, mais en définissant le comportement *macro* du système par exemple à l’aide d’équations différentielles [3, 20, 21]. Cependant, étudier l’émergence de ces phénomènes, et permettre d’interagir avec les abeilles au cours de la simulation, requièrent une approche *bottom-up* pour déduire les effets des actions au niveau des individus (le niveau *micro*) sur la colonie, dans le court et long terme. Découvrir, formaliser et modéliser les liens entre les niveaux micro et macro restent une question scientifique majeure dans les systèmes complexes [32]. Les modèles de population, parce qu’ils se concentrent sur le niveau macro, ne prennent pas

⁽¹⁾L’eusocialité est communément définie comme la participation au bon fonctionnement de la colonie de tous les individus de générations différentes, vivants en communauté.

⁽²⁾Le projet dans lequel s’inscrit ce travail est présenté à l’adresse <https://siaa.univ-brest.fr/simbaca/>

en compte le niveau micro et ses liens existants avec le macro [7]. Nous suivons donc une approche à base d'agents pour modéliser la colonie, dans laquelle une abeille est un agent décrit par son cycle biologique, son comportement, et ses interactions avec son environnement.

Nous présentons dans cet article un modèle à base d'agents des butineuses, et comment leurs comportements, leurs interactions et les mécanismes de recrutement mènent à une sélection des meilleures sources de nectar dans les environs de la colonie. Comme nous nous concentrons sur la ruche pour permettre des actions locales sur les abeilles, comme le retrait d'un cadre (une pratique apicole), nous supposons que l'environnement des agents est la ruche en elle-même, avec la nécessité de définir leur comportement avec une granularité temporelle et spatiale très fine. Cependant, les butineuses vont régulièrement à l'*extérieur de la ruche* pour chercher des ressources (nectar, pollen, eau), dans un rayon pouvant aller jusqu'à 10 km [25], soit une aire d'approximativement 314 km². Il y a donc une grande différence de granularité et d'échelle entre la ruche (habituellement 38 × 45 × 31 cm) et l'environnement extérieur, rendant difficile la connexion entre ces deux environnements.

Nous proposons de mettre en place un module, au sens logiciel du terme, pour adresser cette problématique : le **Module Écosystème** (cf. partie 3). Ce module paramétrable représente de façon symbolique l'environnement extérieur pour abstraire ses dimensions spatiales et temporelles. Il se base pour cela sur des données issues de la météo locale, la longueur de la journée, et les sources de nectar et de pollen environnantes, ainsi que sur des probabilités déduites, adaptées ou empruntées de modèles existants dans la littérature. Le Module Écosystème est compatible avec un modèle à base d'agents : les Agents Butineuses, décrits dans la partie 4, interagissent avec son interface, sous la forme d'un agent, pour simuler *numériquement* la durée de leur sortie de la ruche et les ressources rapportées.

Dans la partie suivante, nous présentons le principe d'auto-organisation des butineuses et ses principaux mécanismes, dont le recrutement, d'un point de vue biologique. Nous passons également en revue plusieurs travaux importants sur la simulation à base d'agents de l'activité de butinage. Deux expérimentations de notre modèle et leurs résultats sont ensuite présentés dans la partie 5. La première expérimentation (partie 5.1), qui avait pour but de *vérifier* le comportement des agents et le rôle du Module Écosystème, a été conduite dans les conditions (météo, sources) de trois années (de 2015 à 2017) à Brest (Bretagne), avec notre modèle et le Module Écosystème. La seconde expérimentation (partie 5.2) a consisté à *valider* le comportement des agents dans la reproduction du phénomène d'auto-organisation que nous cherchions à reproduire. Enfin, nous concluons et présentons les perspectives de ce travail dans la partie 6.

2. AUTO-ORGANISATION ET MODÈLES À BASE D'AGENTS DE BUTINAGE

Cette partie présente tout d'abord les fondements biologiques du butinage et ses mécanismes d'auto-organisation, avant de faire un rapide état de l'art des modèles à base d'agents de butinage existants.

2.1. AUTO-ORGANISATION DES BUTINEUSES

Dans la littérature, les abeilles dont la tâche est de butiner (les butineuses, par abus de langage) se divisent en trois rôles [25] :

- (1) Les « scouts », qui explorent l'environnement pour y trouver de nouvelles ressources (nectar, pollen, eau) ;
- (2) Les « sans-emploi », qui se reposent dans la ruche ou qui sont à la recherche de danses sur la piste de danse ;
- (3) Les « employées » qui exploitent une seule source.

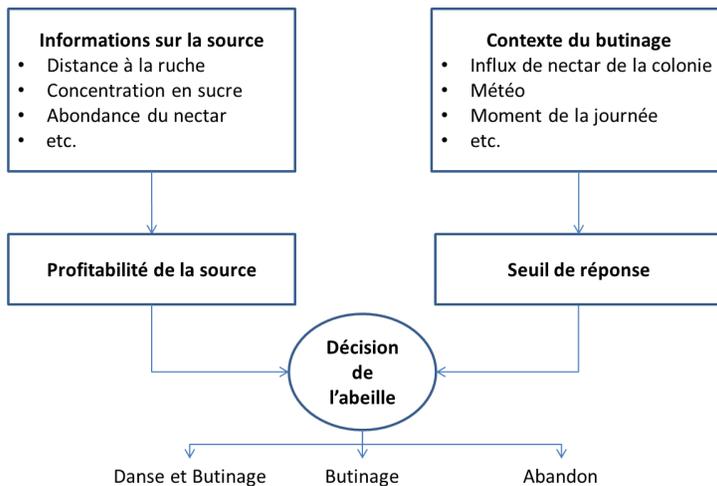


FIGURE 2.1. Diagramme illustrant la prise de décision de l'abeille : danser pour recruter sur sa source avant de continuer à butiner, simplement continuer à butiner, ou abandonner la source. Adapté de Seeley [25].

Les butineuses peuvent passer d'un état à l'autre via le recrutement et l'abandon, selon notamment leur perception de la *profitabilité* de la source, l'influx en nectar de la colonie, et un seuil de réponse pour danser, propre à chaque abeille (voir Figure 2.1). Le recrutement consiste, pour les « scouts » et les « employées », à communiquer à leur retour du butinage, par la danse (*waggle dance*) la position de leur source à des « sans-emploi ». La danse s'effectue à l'intérieur de la ruche, sur les cadres près de l'entrée. Une « scout » ou une « employée » estimant que sa source n'est pas assez profitable pourra continuer d'y butiner sans recruter d'autres butineuses, voire l'abandonner. Dans ce dernier cas, elle deviendra une « sans-emploi ». Seeley [25] a également souligné que les stocks de ressources de la colonie influençaient le comportement des butineuses : lorsqu'ils sont bas, les butineuses ont tendance à explorer plus souvent et à accepter plus facilement des sources peu profitables.

Comme montré par de précédents travaux [23, 26], le recrutement est la pierre angulaire d'un processus d'*auto-organisation* [6] des butineuses, qui les amène à

sélectionner de manière décentralisée les sources de nectar les plus profitables dans les environs de la ruche. Une butineuse qui danse pour recruter ne va pas communiquer directement la profitabilité de la source, mais celle-ci va jouer sur la *durée de la danse* : pour une source très profitable, une butineuse va danser longtemps, augmentant mécaniquement ainsi le nombre d'abeilles qu'elle peut recruter.

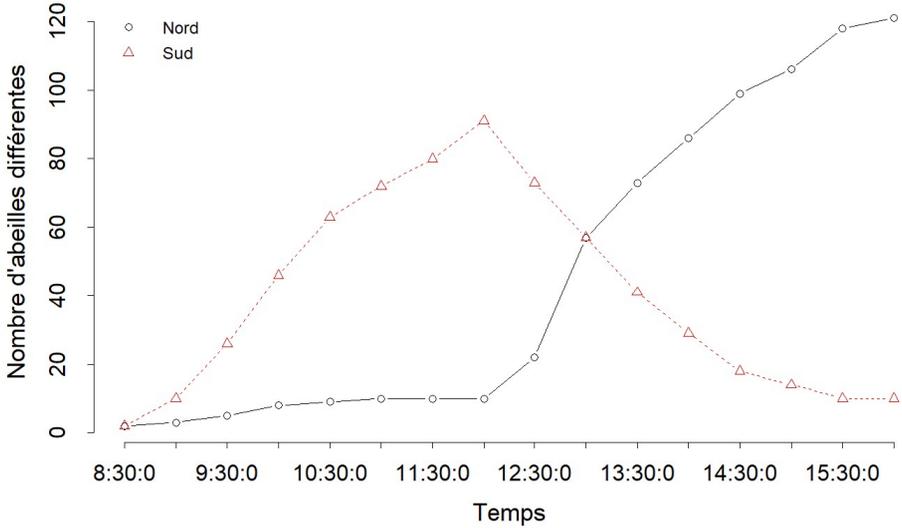


FIGURE 2.2. Résultats adaptés de l'expérience de Seeley et al. [26]. La source Sud commence à 2.5 mol/L de sucre, et la source Nord à 1.0 mol/L. La source Sud est donc la plus profitable. À midi, ces concentrations sont échangées (la source Sud passant même à 0.75 mol/L) et la source Nord devient alors la plus profitable. On peut observer nettement cette cassure sur les courbes, où la source Sud, alors bien exploitée, se voit abandonnée en masse, tandis que l'affluence à la source Nord augmente subitement, montrant l'auto-organisation rapide des butineuses.

Seeley et son équipe [26] ont notamment présenté en 1991 les résultats d'une expérience qui souligne de façon remarquable ce processus d'auto-organisation. Cette expérience consiste à confronter les butineuses d'une colonie à deux sources différentes, placées à égale distance de la ruche (400 m), une au nord et l'autre au sud. Ces sources peuvent être des soucoupes, par exemple, contenant un liquide sucré, dont la teneur en sucre est contrôlable. L'expérience se déroule de 8 h à 16 h, et le nombre de butineuses différentes qui visitent chaque source est enregistré et remis à zéro toutes les demi-heures. Initialement, entre 12 et 15 abeilles sont sélectionnées et entraînées à butiner à chaque source. La concentration en sucre est de 1.0 mol/L pour chacune des sources, ce qui rend les sources juste assez profitables pour encourager ces abeilles à continuer leur butinage mais sans recruter d'autres abeilles. De 8 h à 12 h, la source Sud a une teneur en sucre de 2.5 mol/L, contre 1.0 mol/L pour la source Nord. À 12 h, la teneur en sucre de la source Nord est augmentée à 2.5 mol/L, et celle de la source

Sud est abaissée à 0.75 mol/L. Les résultats de cette expérience sont présentés en Figure 2.2. Ils montrent que l'auto-organisation des butineuses est relativement rapide et autonome, pour toujours allouer la majorité des butineuses sur la source la plus profitable.

2.2. MODÈLES À BASE D'AGENTS DE BUTINAGE

La grande majorité des modèles à base d'agents existants sur le butinage ne modélisent pas l'intérieur de la ruche dans son ensemble, mais tous représentent explicitement l'extérieur de la ruche, le plus souvent sous la forme d'une grille en 2D dans laquelle des sources de pollen et de nectar sont disséminées. Ils complètent parfois cette représentation par des données météorologiques, qui influencent les conditions de sortie des butineuses (cf. partie 3). Ces modèles sont difficilement intégrables avec un modèle spatialement explicite de la ruche, à cause notamment de leur différence d'échelles (temps et espace). La plupart de ces modèles sont indépendants, construits pour étudier une ou plusieurs questions biologiques précises comme, par exemple, les bénéfices du recrutement, du point de vue de l'évolution, selon la taille de la colonie et la qualité des sources [11]. Ces modèles se concentrent sur le comportement de butinage de l'agent hors de la ruche, et introduisent certains paramètres et probabilités intéressants que nous adaptons ici.

Nous pouvons cependant citer deux modèles à base d'agents qui font le lien entre l'extérieur et l'intérieur de la ruche. Le modèle de De Vries et Biesmeijer [33] inclut une représentation 2D de l'environnement de la ruche (1 200 m × 1 200 m) et de la piste de danse (25 cm de rayon), sans que soit spécifié si l'environnement est continu. Le pas de temps de simulation n'est pas indiqué. Si le rayon de perception des agents sur la piste de danse est connu, la durée de leurs danses ne semble pas dépendre de la profitabilité de la source, simplifiée ici et égale à la concentration en sucre des sources. Dans ce modèle, seules les butineuses sont modélisées, mais le rôle de scout n'est pas pris en compte. Le but de ce modèle, implanté en Smalltalk, est de reproduire les résultats de l'expérience de Seeley et al. [26] décrite ci-dessus, ce qu'il parvient à faire avec succès. Cependant, la prise de décision de l'abeille (danser, continuer ou abandonner) ne semble dépendre que de la profitabilité de la source et d'une probabilité d'abandonner, calibrée par expérimentation.

Le modèle présenté par Schmickl et al. [22] est, à notre connaissance, le plus complet à ce jour. Ce modèle, par certains aspects, est proche de celui que nous proposons ici. Par exemple, comme dans le modèle que nous présentons, les agents évaluent la profitabilité des sources qu'ils rencontrent selon un ratio gain-coût énergétique. La durée de leurs danses dépend de cette profitabilité, des stocks de la colonie et de leur fonction de stimulus-réponses, propre à chaque individu. Ces mécanismes sont décrits dans la partie 4.2.

La première différence notable entre ce modèle et celui que nous proposons ici réside dans sa représentation de l'environnement de la ruche. Chez Schmickl et al. [22], l'environnement est continu, et consiste en une représentation 2D de l'extérieur de la ruche et de la piste de danse. Le pas de temps de simulation est de 0.5 seconde, à

l'intérieur comme à l'extérieur de la ruche. Les données météorologiques ne sont pas prises en compte, et au maximum trois sources « artificielles », sans floraison, peuvent être représentées. Pour ces raisons, notamment, ce modèle est utilisé pour simuler seulement quelques heures de butinage (dont l'expérience de Seeley et al. [26]).

Une autre différence notable est que le modèle de Schmickl et al. [22] s'intéresse en particulier à la relation et au ratio entre les butineuses et les receveuses (dont le rôle est de récupérer le nectar que rapporte les butineuses pour le stocker dans la ruche), et à leur impact sur les résultats du butinage. En plus de la *waggle dance*, il prend donc en compte la *tremble dance*, qui est liée directement à l'influx de nectar dans la colonie et a pour effet de recruter plus de receveuses. La prise de décision de l'agent (danser, continuer à butiner, abandonner) est donc différente. Notre modèle se concentre seulement sur les butineuses et la nature des mécanismes de recrutement dans leur auto-organisation, indépendamment des receveuses.

Enfin, le modèle de Schmickl et al. [22] est implanté en NetLogo [34], limitant son extensibilité et les perspectives de mettre en place les interactions naturelles avec la ruche, ce qui n'est pas le cas avec notre proposition.

Dans la littérature, on trouve également quelques travaux couplant des modèles à base d'agents et des modèles de population dans le but de simuler la dynamique de la colonie, les phénomènes complexes n'étant pas pris en compte. En simulation, le couplage de modèles consiste à utiliser des modèles basés habituellement sur des paradigmes différents et interagissant, soit pour représenter un même phénomène à différentes échelles, soit pour modéliser plusieurs phénomènes [14]. C'est le cas par exemple du modèle BEEHAVE [3], dont le modèle à base d'agents dédié au butinage interagit avec deux modèles de population simulant la vie de la colonie (naissances, consommation des ressources, etc.) et la propagation du parasite *Varroa destructor* dans la colonie. Plusieurs propriétés du modèle rendent difficile son adaptation à un modèle à base d'agents de la colonie : les agents butineuses sont des « super-individus » représentant 100 butineuses identiques, et le modèle est exécuté une fois par jour, avec un pas de temps de la simulation qui semble être la minute. De la même manière, dans Bee++ [5], le couvain est simulé au niveau populationnel, l'intérieur de la ruche n'étant pas modélisé. Le comportement des agents est de plus guidé par des données globales (le nombre courant d'agents en train de butiner, le nombre total d'agents dans la colonie, etc.) qui sont imperceptibles, biologiquement parlant, par les abeilles.

Pour terminer, il peut être intéressant de mentionner qu'une formalisation logique d'une partie de ces mécanismes de recrutement a été proposée par Sumpter et Broomhead [28], et donne un point de vue intéressant sur le comportement des butineuses, mais ne montre pas comment le processus d'auto-organisation des butineuses se met en place.

Pour résumer, le modèle que nous présentons propose de s'affranchir des problèmes d'échelles posés par les différents environnements des agents, pour se concentrer sur l'intérieur de la ruche et les phénomènes complexes qui s'y trouvent. Nous nous intéressons plus particulièrement aux différents mécanismes nécessaires à l'auto-organisation des butineuses.

3. MODULE ÉCOSYSTÈME

Le rôle du Module Écosystème est de simuler, à la demande d’un agent, l’activité de butinage en se basant sur une représentation *symbolique* et fonctionnelle de l’environnement extérieur de la ruche. Cette représentation inclut deux facteurs importants : des données sur la météo locale, et les sources de pollen et de nectar environnantes, avec leurs périodes d’activité. Ces deux facteurs font chacun l’objet d’un module dédié et paramétrable, les modules Climat et Source, tandis qu’un dernier composant consiste en un agent qui sert d’interface.

Les conditions de sortie de la ruche des butineuses sont influencées [25] par :

- (1) la longueur de la journée, les abeilles ne butinant que le jour ;
- (2) la météo, les butineuses sortant de la ruche lorsque la température extérieure est aux alentours de 11°C, et qu’il n’y a pas de pluie ou de vent fort.

Le module Climat modélise ces aspects en calculant les heures de lever et de coucher du soleil, et en se basant sur les données réelles de météo, suivant la date et la localisation de la simulation, qui sont paramétrables. Les données météo (température, humidité, nébulosité, etc.) proviennent du service de données publiques de Météo France ⁽³⁾, et sont disponibles pour soixante-deux stations en France. À partir de ces mesures et du moment de la journée, le module Climat peut donc indiquer si une butineuse peut ou non sortir de la ruche.

Le module Source permet de représenter les sources de nectar et de pollen environnantes, leur distance à la ruche, la période, durée et pic de floraison, ainsi que la concentration en sucre du nectar disponible. Lorsqu’une simulation est lancée, on peut choisir de générer un certain nombre de sources aléatoires, mais aussi de configurer les principales sources locales et le nombre d’occurrences de ces sources. Par exemple, certaines des sources connues autour de Brest, dans le Finistère, et utilisées dans notre expérimentation (voir partie 5.1), sont : le colza (15 avril au 15 mai), les ronces (15 juin au 15 juillet), le châtaignier, le saule, etc.

Il est également possible de configurer des sources artificielles, dont on peut régler la concentration en sucre en cours de simulation, ce qui permet de se rapprocher des protocoles expérimentaux *in-vivo* et de reproduire ces expériences (voir la partie 5.2).

Chaque source peut être régulée par un *facteur source*, directement inspiré du « season factor » du modèle de population HoPoMo [21], et qui simule les périodes de floraison, c’est-à-dire la disponibilité des ressources de nectar et/ou de pollen selon la date. Il est défini par la fonction Gaussienne normalisée :

$$SF(src, nb) = 1 - \left(\frac{1}{\frac{d}{4} \cdot \sqrt{2} \cdot \Pi} \cdot \exp \left(- \frac{(nb - p)^2}{2 \cdot \frac{d^2}{4}} \right) \right)$$

où *src* est la source, *nb* le numéro du jour de l’année, *d* et *p* la durée et le pic de floraison.

⁽³⁾<https://donneespubliques.meteofrance.fr/>

Chaque source a également une *profitabilité*, perçue par chaque butineuse découvrant la source, qui dépend notamment de sa distance à la ruche, de la concentration en sucre du nectar disponible, et de la quantité de ressources ramenées (voir partie 4.2).

Le dernier composant du Module Écosystème est l'*Agent Écosystème*, qui est l'interface du système. Cette interface est un agent, permettant ainsi à une simulation à base d'agents de communiquer avec le Module Écosystème, via un protocole de communication respectant le standard FIPA-ACL [15]. Typiquement, une butineuse de notre modèle (que nous appellerons **Agent Butineuse**) demande (*Request*) à l'Agent Écosystème s'il peut aller butiner à une source connue (soit par recrutement, soit qu'il a déjà visitée) ou à une source qu'il doit d'abord découvrir. Selon les probabilités p_{mort} et $p_{succès}$ (cf. Table 6.1), l'Agent Écosystème répond par un *refus* à la requête (parce que l'Agent Butineuse meurt, ou n'a trouvé aucune source) ou *accepte* la requête. Dans ce dernier cas, l'Agent Écosystème communique à l'Agent Butineuse (1) la quantité de nectar et/ou de pollen récoltée, (2) la qualité du nectar de la source et (3) le temps nécessaire au butinage, exprimé en nombre de pas de simulation (voir la partie 4.4) que l'Agent Butineuse doit attendre pour simuler sa sortie.

4. MODÈLE MULTI-AGENT DE BUTINAGE

Nous ne considérons dans ce travail préliminaire que les butineuses. Les autres abeilles et les interactions qu'elles peuvent avoir avec les butineuses ne sont pas prises en compte pour le moment. Un autre modèle, complémentaire à celui présenté ici, se concentre sur l'allocation des tâches dans la colonie [1], et contribue à notre objectif de représenter l'ensemble de la colonie. À partir de la littérature et de l'expertise de nos partenaires biologistes, nous modélisons donc le cycle biologique des butineuses, leur comportement, et leurs interactions avec leur environnement.

4.1. ENVIRONNEMENT

L'environnement d'un Agent Butineuse est constitué de la ruche, des autres agents et de l'Agent Écosystème, comme illustré par la Table 4.1. Ce type de table est similaire aux matrices d'interactions introduites par le modèle IODA [18], et couramment utilisé dans une approche de modélisation orientée interactions (à l'opposé d'une approche orientée agents). Elle participe ici à la documentation de notre modèle.

La représentation que nous avons choisie de la ruche contient et met à jour les valeurs des stocks de nectar et de pollen. L'influx de nectar de la colonie qui influence l'état des stocks de la colonie est donc inclut dans cette représentation. Nous supposons que cette représentation est pour le moment suffisante pour reproduire l'auto-organisation des butineuses, mais notre objectif à court terme est d'avoir une représentation spatiale explicite de la ruche, et notamment de la piste de danse.

Ce choix de conception a une première conséquence portant sur les stocks de la colonie. La consommation des stocks de ressources par les abeilles est simulée à partir du nombre d'abeilles total de la colonie, selon les proportions de chaque spécialité dans

TABLE 4.1. Table d’interactions de l’Agent Butineuse avec son environnement.

Interactions	Agent Butineuse	Ruche	Agent Ecosystème
Agent Butineuse	Recrutement	Stockage des ressources	Envoi de la requête de butinage
Ruche	Communication des stocks	Consommation des ressources	-
Agent Ecosystème	Accepter / Refuser la requête de butinage	-	-

la littérature. Cela signifie que les stocks peuvent diminuer lorsqu’il n’y a pas d’entrée de pollen ou de nectar. De plus, dans une colonie, la perception par les abeilles de ces stocks est un phénomène complexe, qui implique notamment ses interactions avec d’autres abeilles (les receveuses). Dans ce modèle, la perception est donc abstraite des valeurs de stocks perçues au travers d’un biais aléatoire pour simuler son imprécision.

L’état des stocks de ressources va influencer le comportement de l’Agent Butineuse, notamment en pondérant sa prise de décision de recruter ou non, mais aussi les probabilités de partir de lui-même explorer (p_{scout}) ou d’aller sur la piste de danse (p_{piste_danse}). Cette influence est décrite ci-après. La Table 6.1 située en Annexes donne les valeurs de toutes les probabilités et paramètres utilisés dans le modèle.

Ce choix de conception a une seconde conséquence, portant cette fois sur la danse. La piste de danse n’ayant pas pour le moment de représentation spatiale, les Agents Butineuses « sans-emploi » simulent leur recherche de danse sur la piste par un temps d’attente aléatoire (paramètres T_{Min} et T_{Max}), à la suite duquel ils recherchent les agents en train de danser.

4.2. MODÉLISATION DU RECRUTEMENT

Nous avons vu dans la partie 2 que le recrutement était lié à la rentabilité de la source, et à la décision de l’abeille de danser, (et si oui, comment), de continuer à butiner ou d’abandonner cette source. Comme présenté dans la partie 5.2, les résultats montrent que les mécanismes décrits ci-dessous aident à reproduire l’auto-organisation des butineuses dans les simulations.

4.2.1. Perception de la rentabilité des sources

Seeley a montré que les butineuses cherchent à maximiser l’efficacité énergétique nette du butinage [24]. Lorsqu’elles collectent G unités d’énergie sur une source, tout en dépensant C unités d’énergie pour cette collecte, cette source est d’autant profitable que $\frac{G-C}{C}$ est grand. Une source profitable est donc une source qui donne le ratio maximum gain énergétique / coût énergétique.

Dans notre modèle, chaque Agent Butineuse évalue la rentabilité d’une source selon cette équation. Pour modéliser le gain G , nous suivons la définition donnée par

Seeley [24] avec $G = V \times S \times 5.8$, où V est le volume de nectar ramené par l'Agent Butineuse en μL et S la concentration en sucre de la solution en mol/L. Le calcul du coût énergétique C est lui simplifié, et est fonction de la distance D de la ruche jusqu'à la source et de la consommation moyenne de nectar E en mg/m de l'Agent Butineuse en vol : $C = D \times E \times 9.5$.

4.2.2. Décision de l'Agent Butineuse : recrutement et/ou butinage, ou abandon de la source ?

Une fois la profitabilité de la source évaluée par l'Agent Butineuse, celui-ci doit prendre une décision : cette profitabilité est-elle suffisamment importante pour qu'il communique la source à d'autres agents ? Pour qu'il continue seulement à y butiner ? Ou doit-il l'abandonner ?

Comme le modèle présenté par Schmickl et al. [22], nous nous basons sur des fonctions de stimulus-réponses introduites par Seeley [24], qui sont des fonctions affines de type $y = a \times x - b$, avec $a \in [0; 1.5]$ et $b \in [3; 20]$. Ces fonctions, propres à la génétique de l'individu, font le lien entre la profitabilité perçue par l'individu et au nombre de mouvements (*waggle*) à effectuer au cours de sa danse – et donc à la durée de la danse – s'il décide de recruter. La Figure 4.1 montre un exemple de diversité de fonctions que l'on peut obtenir dans notre modèle. Les trois profils plus sombres et hachurés sont des profils d'agents que nous sélectionnons lors de la seconde expérimentation, que nous décrivons en partie 5.2. En dehors de cette expérimentation, par défaut dans le modèle, chaque Agent Butineuse est créé avec son propre profil génétique, en tirant aléatoirement selon une loi uniforme les valeurs de a et de b dans leur intervalle.

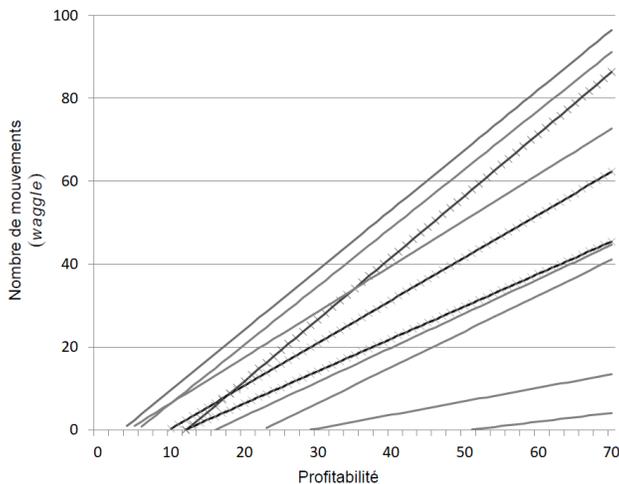


FIGURE 4.1. Dix profils générés aléatoirement lors d'une instanciation de notre modèle. Les profils sont sous la forme d'une droite $a \times x - b$.

À chaque fonction de stimulus-réponses est associé un seuil s au-delà duquel l'Agent Butineuse va danser, donné par l'équation $s = \frac{b}{a}$. Suivant Seeley [25, 24], l'état des stocks de la colonie va venir influencer la prise de décision de l'Agent en le poussant à accepter des sources moins profitables lorsque les stocks sont bas, et à être plus sélectif dans le cas contraire. Dans notre modèle, l'état des stocks est représenté par un nombre réel entre $[0; 1]$ qui influence la valeur du paramètre b de la fonction de stimulus-réponses : plus les stocks sont hauts, plus le seuil impacté par ces stocks s_{impact} est haut, et inversement.

Nous proposons donc ici les règles suivantes pour la prise de décision de l'Agent Butineuse :

- (1) Si la profitabilité est supérieure au seuil s_{impact} , alors il danse ;
- (2) Si la profitabilité est inférieure au seuil s_{impact} , mais supérieure au seuil s , alors il continue à butiner sur la source mais ne danse pas ;
- (3) Si la profitabilité est inférieure au seuil s_{impact} et inférieure au seuil s , alors il abandonne la source.

Pour le dire plus simplement, lorsque les stocks sont critiques, s_{impact} sera confondu avec le seuil s , poussant les Agents à toujours danser dès qu'une source est suffisamment profitable.

4.2.3. Danse et recrutement

La fonction de stimulus-réponses de l'Agent Butineuse lui donne le nombre de mouvements à effectuer au cours de sa danse, si la profitabilité de la source est supérieure à s_{impact} . La durée d'un mouvement dépend directement de la distance de la source à la ruche (75 ms tous les 100 m), et la danse en elle-même est séparée en plusieurs périodes (entre deux et cinq) entre lesquelles l'abeille se déplace sur la piste de danse [25].

À partir de ces données, l'Agent Butineuse calcule donc la durée de la danse qu'il va effectuer. Pendant qu'il danse, il est à l'écoute de messages provenant d'agents « sans-emploi » à la recherche de danses, à qui il va pouvoir communiquer :

- l'identifiant de la source (l'environnement étant représenté de manière symbolique dans le Module Ecosystème) ;
- un temps minimum et maximum aléatoire qui simulent l'écoute de la danse par l'agent « sans-emploi » (paramètres W_{Min} et W_{Max}).

Les agents ainsi recrutés suivent donc tout d'abord une partie de la danse, puis vont contacter l'Agent Ecosystème pour tenter de trouver la source communiquée.

Dans un souci de réalisme, un agent recruté trouve la source en fonction d'une probabilité $p_{succès}$, qui dépend du nombre d'essais qu'il a déjà effectué pour la trouver [25]. Lorsqu'il ne trouve pas la source, il retourne dans la ruche pour rechercher de nouvelles danses. S'il est recruté, plus tard, sur la même source, la probabilité $p_{succès}$ est augmentée.

4.3. COMPORTEMENT DE L'AGENT BUTINEUSE

Grâce à sa représentation graphique simplifiée, le diagramme d'états présenté dans la Figure 4.2 a été le support d'une discussion interdisciplinaire entre biologistes, apiculteurs, et informaticiens, en vue de définir le comportement de l'Agent Butineuse. Il offre une vue dynamique du comportement de l'agent dans le modèle.

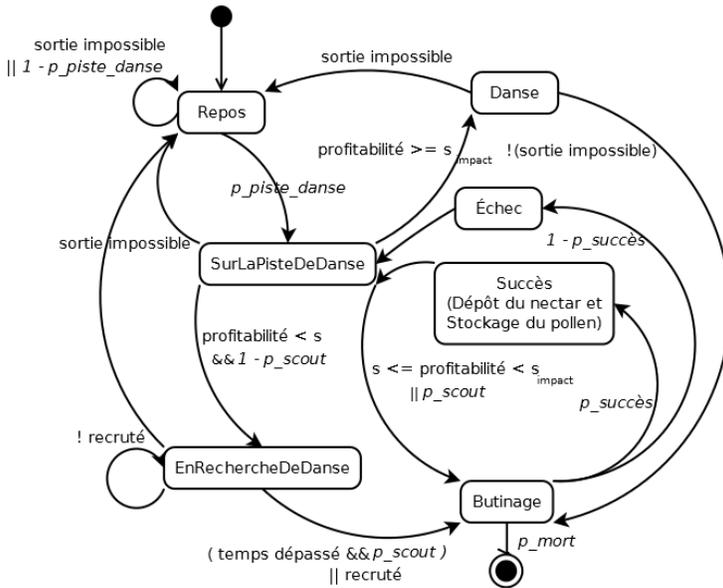


FIGURE 4.2. Diagramme d'états du comportement de l'Agent Butineuse.

Dans la Figure 4.2, les transitions entre les états sont des probabilités et/ou des conditions vérifiées par l'agent lorsque la ou les actions dans chaque état sont terminées. La Table 6.1, située en Annexes, complète le modèle et donne les valeurs des probabilités utilisées par l'Agent Ecosystème et l'Agent Butineuse, ainsi que les valeurs des différents paramètres, qui peuvent être légèrement différentes pour chaque abeille pour représenter une variabilité inter-individuelle. La calibration de certains paramètres a été effectuée grâce à l'étude de leur influence, présentée en partie 5.2.2.

L'Agent Butineuse suit donc le comportement défini par le diagramme d'états, constitué de sept états. Les états les plus importants sont :

SurLaPisteDeDanse : cet état est au carrefour de plusieurs transitions. L'agent peut venir de l'état *Repos*, pour soit attendre d'être recruté par un autre agent, soit sortir explorer (probabilité p_{scout}). L'agent peut aussi venir de l'extérieur, en ramenant ou non des ressources, et décider de recruter (*Danse*), continuer de butiner (*Butinage*) ou abandonner sa source (retour à l'état *EnRechercheDeDanse*) en fonction de sa profitabilité.

EnRechercheDeDanse : toutes les t secondes (t tirée aléatoirement entre T_{Min} et T_{Max}), l'agent recherche par envoi de messages d'autres agents dans l'état *Danse*. Dans le cas où il reçoit une réponse positive, l'agent suit la danse pendant un certain temps ($\in [W_{Min}; W_{Max}]$) puis passe dans l'état *Butinage*. Dans les autres cas, et si l'agent estime avoir attendu trop longtemps, il peut également sortir explorer suivant la probabilité p_scout .

Butinage : cet état simule le butinage en lui-même, via la communication avec l'Agent Écosystème. *Danse* : après une sortie fructueuse, l'agent revient sur la piste de danse et peut essayer de recruter d'autres agents dans l'état *EnRechercheDeDanse*, suivant la profitabilité de la source et sa prise de décision décrite précédemment (partie 4.2).

La prise en compte de la temporalité est faite à différents niveaux, dans le modèle de comportement de l'agent et dans le Module Écosystème. La durée totale du butinage est ainsi calculée par le Module Écosystème à partir de la distance à la source, la vitesse de vol et le temps passé sur la source, quand elle est trouvée. La durée de la danse est elle calculée à partir de la profitabilité de la source (cf. partie 4.2.3). Ces durées (en secondes) se traduisent ensuite en autant d'*actions* unitaires d'une seconde que nécessaire, comme détaillé dans la section suivante. Enfin, certaines probabilités utilisées dans les transitions d'états expriment également une certaine durée : la valeur de p_piste_danse , par exemple, indique que les chances pour un agent de passer de l'état *Repos* à l'état *SurLaPisteDeDanse* sont relativement faibles, et donc qu'en moyenne les agents se reposent un certain temps.

4.4. IMPLANTATION ET ORDONNANCEMENT

Le modèle à base d'agents et le Module Écosystème ont été implantés avec JADE [4] dans le langage de programmation orienté-objet Java. JADE intègre notamment des mécanismes de communication, et une architecture adaptée à notre modèle et à son évolution : les comportements sont indépendants de l'agent, et JADE propose notamment une classe *FSMBehavior* permettant d'exprimer directement le diagramme d'états et ses transitions. Ce choix de plateforme repose aussi sur la nécessité d'implanter un modèle dans un langage de programmation standard (*i.e.* à l'inverse de langages spécifiques comme celui de NetLogo [34], par exemple), pour rendre le code plus réutilisable, extensible et distribuible.

Par défaut, à un pas de temps de la simulation (`time_step`) correspond une seconde en réalité. Par défaut également, la vitesse de la simulation est de 1 000 : cela signifie que les *actions* qui prennent une seconde à une abeille réelle, prennent une milliseconde à nos Agents Butineuses dans la simulation. Ces *actions* ont été définies comme telles, en accord avec la biologie, à l'intérieur des états du comportement de l'Agent Butineuse (cf. Figure 4.2).

Le simulateur utilise deux générateurs de nombres aléatoires, dont les graines peuvent être paramétrées : un générateur principal pour les Agents (profils de stimulus-réponse, probabilités etc.) et un générateur qui a trait exclusivement aux

sources (quantités de nectar / pollen récupérées, durée du butinage etc.). La mise en place de ces deux générateurs séparés nous permet par exemple de tester l'influence des profils des Agents, en ne modifiant que la graine du générateur lié aux sources.

Un agent en charge de l'ordonnancement, l'Agent Ordonnanceur, a deux politiques d'activation des agents de la simulation suivant que l'on veuille simuler sur une période courte (*e.g.* 24 h ou 48 h) ou plus longue (plusieurs mois). Dans les deux cas, c'est un ordonnancement asynchrone que nous avons choisi, dans lequel les agents sont activés dans un ordre aléatoire pour limiter les biais [17]. La simulation sur une courte période repose sur une politique d'ordonnancement classique, décrite ici en dans le formalisme du pseudocode proposé par Drogoul et al. [13] :

```
1  while(globalVirtualTime != endOfSimulation){
2      shuffle(AllTheAgents);
3      send(activationMsg, AllTheAgents.get(0));
4
5      while(receiveMsg() == null)
6          wait();
7          // Attente passive (communication dans JADE)
8
9      virtualEnvironment.evolve();
10     globalVirtualTime++;
    }
```

L'Agent Ordonnanceur envoie un message d'activation au premier agent de sa liste (différent à chaque pas de temps), qui fait une *action* puis transmet le message au second agent dans la liste, et ainsi de suite. Le dernier agent de la liste répondra lui à l'Agent Ordonnanceur après son exécution. L'Agent Ordonnanceur fait ensuite évoluer l'environnement, et avancer le temps de la simulation. Cette politique d'ordonnancement a notamment l'avantage de baser l'aléatoire de la simulation sur les graines des deux générateurs, et ainsi reproduire exactement les résultats obtenus dans une simulation. Cependant, si elle est adaptée à des simulations sur des périodes courtes, elle prend beaucoup de temps sur de plus longues périodes : à titre d'exemples, la simulation de 8 heures de butinage avec 200 abeilles (expérience décrite en 5.2) prend une dizaine de minutes, tandis que la simulation de 6 mois de butinage, avec 500 abeilles (expérience décrite en 5.1) prend 138 heures.

Pour simuler sur une longue période, l'Agent Ordonnanceur envoie un message à *tous* les agents, avec un nombre d'actions *nbActions* à effectuer.

```
1  while(globalVirtualTime != endOfSimulation){
2      nbAnswers=0; // Nombre de réponses reçues
3      nbEvolution=0;
4      // Nombre de fois que l'environnement et le temps ont évolué
5
6      shuffle(AllTheAgents);
7      for (SimulatedAgent agent : AllTheAgents)
8          send(activationMsg, agent);
```

```

9           // Envoie d'un message à tous les agents
10
11         while(nbAnswers != AllTheAgents.size()){
12           // Attente de toutes les réponses
13           while(receiveMsg() == null){
14             wait(nbActions x time_step);
15             // Attente passive limitée dans le temps
16
17
18             if(nbEvolution < n){
19               virtualEnvironment.evolve();
20               globalVirtualTime++;
21               nbEvolution++;
22             }
23           }
24           nbAnswers++;
25         }
26
27         for(int i=0;i< n - nbEvolution;i++){
28           virtualEnvironment.evolve();
29           globalVirtualTime++;
30         }
31       }

```

À la réception du message, les agents redeviennent des threads indépendants et donc s'exécutent lors du temps d'exécution qui leur est alloué par la machine virtuelle Java (JVM). Lorsqu'un agent a terminé ses *nbActions* actions, il informe l'Agent Ordonnanceur qu'il a terminé, et attend à nouveau un message d'activation. Cette synchronisation des agents s'inspire des mécanismes décrits dans Braubach et al. [9] et Cich et al. [10], et a pour but de réduire le biais lié à l'ordonnement de la JVM, qui peut favoriser certains agents en leur donnant plus de temps d'exécution qu'à d'autres.

Une des différences avec le pseudocode précédent se situe aux lignes 13-22 : au lieu d'attendre les réponses de tous les Agents Butineuses *puis* de faire évoluer *nbActions* fois l'environnement et le temps de la simulation, l'Agent Ordonnanceur fait cette évolution *pendant* l'exécution des Agents Butineuses, après avoir attendu un certain temps, fonction de *nbActions* et du pas de temps de la simulation *time_step*. L'idée ici est d'éviter que les Agents Butineuses ne fassent leurs *nbActions* actions sur le même état de l'environnement. Les lignes 27-30 sont là pour s'assurer qu'une fois toutes les réponses reçues, l'environnement et le temps ont bien été mis à jour *nbActions* fois.

Avec cette politique d'ordonnement, plus la valeur de *nbActions* est petite, plus le nombre de messages échangés sera grand. À l'inverse, une trop grande valeur pour *nbActions* peut augmenter l'écart en nombre d'*actions* effectuées entre les agents. Cette seconde politique nous permet de gagner du temps, ce qui est intéressant pour des simulations de longues périodes : sur l'exemple donné ci-dessus, la simulation de 6 mois de butinage avec 500 abeilles prend 72 heures

avec $nbActions = 60$, soit un gain de plus de la moitié. Le problème posé par cet ordonnancement est celui de la reproductibilité des expérimentations : en raison de l'aléatoire imposé par l'ordonnanceur propre à Java, il ne sera pas possible, à graine égale, de reproduire de façon fidèle la même simulation. N'utiliser qu'un seul thread, à la manière de la politique d'ordonnancement précédente, réglerait ce problème, mais au prix de la rapidité obtenue.

5. EXPÉRIMENTATIONS ET RÉSULTATS

Deux expérimentations de notre modèle ont été menées : une expérimentation sur une période longue, et une autre sur une journée. L'expérimentation sur le long terme avait pour but de *vérifier* la dynamique du système et l'influence du Module Écosystème, tandis que celle sur une journée a consisté à *valider* le comportement des agents dans la reproduction du phénomène d'auto-organisation que nous cherchions à reproduire.

5.1. EXPÉRIMENTATION SUR LA PÉRIODE LONGUE

Dans cette première expérimentation, nous avons conduit trois expériences avec notre modèle et le Module Écosystème, chacune dans les conditions (météo et sources locales) des années 2015, 2016 et 2017 à Brest. Ces expériences courent du 1er mars au 1er septembre de chaque année. Chaque expérience consiste en 5 essais avec la même météo et localisation des sources, et le même état initial : 500 Agents Butineuses, et les stocks de ressources à un niveau classique en apiculture (état des stocks = 0.4). À chaque essai, la graine du générateur aléatoire est différente. Comme le modèle à base d'agents n'inclut pas pour le moment la reine et les larves, la probabilité p_mort de l'Agent Écosystème a été fixée à 0. La seconde politique d'ordonnancement est utilisée avec $nbActions = 60$.

À partir des résultats de ces trois expériences, nous vérifions notamment que la météo influence la sortie des Agents Butineuses, (cf. partie 3), et que les agents « scouts » représentent en moyenne 10 % du nombre total d'agents [25].

La Figure 5.1 présente les résultats d'une seule exécution sur l'année 2015, avec une graine égale à 1. Elle montre tout d'abord l'influence de la météo sur les agents. Les premiers points sur la gauche du graphe (b) signifient qu'il y a quelques agents sortant de la ruche aux alentours de 12 h (x), mais que la majorité va butiner autour de 17 h (o), jusqu'à 19 h (+). Au mois de mars, il y a plusieurs jours où les agents ne peuvent pas sortir de la ruche à cause de faibles températures (graphe (a)) ou d'une mauvaise météo. À l'inverse (e.g. en juin), les agents peuvent sortir butiner pour la première fois très tôt dans la journée, dès le lever du soleil, et terminer au coucher.

Les effets du mécanisme de recrutement sont illustrés par le graphe (c), qui montre une corrélation entre le nombre d'Agents Butineuses « scouts » (ligne pleine) et recrutés (tirets et pointillés). Au début du mois d'avril, par exemple, une

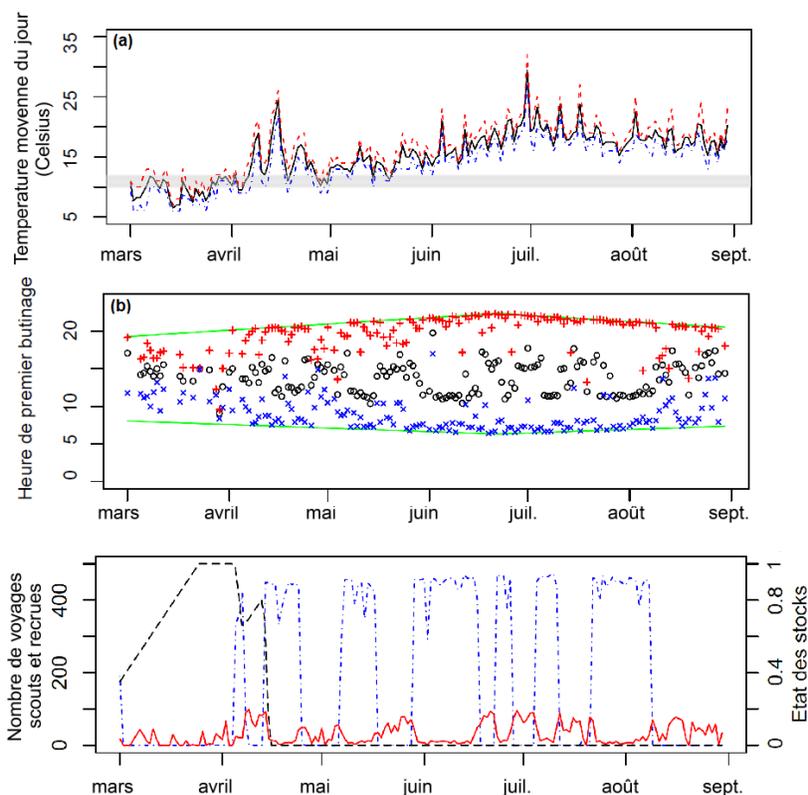


FIGURE 5.1. Le graphe (a) montre la température moyenne par jour de l'année 2015 à Brest (ligne pleine), avec les minimums (tirets et pointillés) et maximums (tirets). La zone grisée correspond à l'intervalle de températures à partir duquel les agents peuvent sortir butiner. Le graphe (b) donne, en moyenne et par jour, l'heure à laquelle des agents sortent butiner pour la première fois (cercles). Les minimums (x) et maximums (+) de ces heures sont également affichés, ainsi que les heures de lever et de coucher du soleil (lignes pleines). Le graphe (c) montre le nombre total de voyages effectués par jour par les agents « scouts » (ligne pleine) et les agents recrutés (tirets et pointillés), ainsi que l'évolution de l'état des stocks (tirets) selon l'axe de droite.

nouvelle source profitable est découverte par les « scouts », et, la météo permettant la sortie de la ruche, un grand nombre d'Agents Butineuses est recruté en quelques heures vers cette source ; presque 50 % du nombre total d'agents. Ce phénomène s'explique par l'état des stocks de la colonie indiquant l'urgence (état = 1), et qui influence les agents à aller sur la piste de danse et à accepter des sources potentiellement peu profitables. Le nombre d'agents recrutés augmente donc et seuls quelques « scouts » restent. Ces résultats se retrouvent d'une manière globale dans les essais des trois expériences.

Sur les trois expériences, la proportion moyenne d'Agents Butineuses « scouts » par rapport au nombre total d'agents se situe entre 6 % et 7 %. Cela peut provenir de l'évolution des stocks, ou du calcul de l'état, qui atteint très tôt sur le graphe (c), à partir du mois de mai, une valeur égale à 0, et qui y reste. D'autres expériences, en faisant par exemple varier la valeur initiale des stocks pour quantifier leur impact, sont nécessaires et feront l'objet de prochaines études.

5.2. EXPÉRIENCE DE SEELEY : LE RÔLE DE L'AUTO-ORGANISATION

Cette seconde expérimentation vise à reproduire avec notre modèle l'expérience publiée en 1991 par Seeley et al. [26], décrite dans la partie 2, qui illustre l'auto-organisation des butineuses et leur capacité à sélectionner les sources les plus profitables autour de la ruche. Pour ces expérimentations nous nous sommes rapprochés au mieux des conditions de Seeley. Nous avons placé deux sources à 400 m de la ruche, fixé leur teneur en sucre comme indiqué par Seeley (respectivement 2.5 mol/L et 1.0 mol/L de sucre, puis 0.75 mol/L et 2.5 mol/L après permutation). La permutation se fait également à 12h, et nous relevons le nombre d'abeilles différentes venant butiner sur chaque source toutes les 30 minutes. La météo est celle d'une journée du mois de juin, idéale pour le butinage, et similaire à celle de l'expérience initiale.

La sélection et l'entraînement des abeilles initiales se fait au lancement de chaque simulation en sélectionnant les Agents Butineuses dont le profil, avec la profitabilité de la source à 1.0 mol/L de glucose, les poussent à butiner à cette source mais sans recruter d'autres Agents (un exemple de tels profils, est donnée Figure 4.1 où ils sont hachurés). Ces Agents sont ensuite envoyés sur la source Nord ou Sud, et leur probabilité de succès $p_{succès}$ fixée au maximum (cf. Table 6.1). Pour que de tels Agents Butineuses soient trouvés, il est nécessaire que la colonie ait un bon état des stocks de ressources, entre 0.1 et 0.5. Si les stocks étaient plus critiques, la différence entre le seuil impacté par ces stocks s_{impact} et le seuil s des profils serait trop faible, les Agents préférant ainsi recruter d'autres Agents même si la profitabilité de la source est basse.

Pour vérifier que le phénomène d'auto-organisation des butineuses a lieu dans notre modèle, nous comparons les courbes obtenues par la simulation avec les courbes produites par Seeley. Pour cela, nous introduisons deux indices qui vont nous permettre de mesurer l'écart entre les courbes. Le premier indice mesure l'écart absolu, en calculant la moyenne de la somme des carrés des différences point à point entre les courbes : la *MSE*, ou *Mean Squared Error*. Le deuxième indice, que nous appelons ici *NMSE*, est calculé à partir d'une normalisation de la *MSE*, et mesure l'écart dans la forme des courbes.

La Table 5.1 présente les principaux paramètres de l'expérimentation, avec leur valeur par défaut et l'intervalle utilisé pour explorer leur influence. Pour chaque valeur, 5 simulations ont été effectuées (soit approximativement 50 minutes), avec la même graine pour le générateur aléatoire lié aux Agents (même profils) et une graine différente pour le générateur lié aux sources.

TABLE 5.1. Les principaux paramètres de l'expérimentation, avec leur intervalle de test pour évaluer leur influence.

Paramètres	Valeur	Intervalle de test
Nombre d'abeilles	200	[50; 500]
p_{piste_danse}	0.00005	[0.001; 0.00001]
Influence des stocks	0.2	[0.1; 0.45]
$p_{succès}$:		
Pour l'Agent Butineuse recruté		
- trouve la source au 1 ^{er} essai	0.2	[0.1; 1.0]
- trouve la source au 2 ^e essai	0.4	[0.2; 1.0]
- trouve la source au 3 ^e essai	0.6	[0.3; 1.0]
- trouve la source au 4 ^e essai	0.8	[0.5; 1.0]
- au-delà, et Agent entraîné	0.93	[0.5; 1.0]
Temps d'attente entre deux recherches de danse		
- T_{Min}	60 s	[30; 150]
- T_{Max}	240 s	[160; 500]

5.2.1. Discussion des résultats

La Figure 5.2 montre les courbes obtenues par Seeley et les courbes obtenues avec les paramètres par défaut sur les sources Nord et Sud, moyenne obtenue sur 5 simulations avec la graine 0 du générateur aléatoire principal, et en faisant varier la graine du générateur lié aux sources. Nous pouvons observer que, comme pour les courbes de Seeley, un fort recrutement est effectué pour la source Sud avant midi. Un fort recrutement est de même effectué vers la source Nord après midi, une fois que les concentrations en sucre des sources ont été échangées et que cette source Nord est désormais plus profitable. La source Sud est abandonnée massivement juste après la permutation. Nous avons donc bien une auto-organisation des butineuses dans une même journée pour la sélection des meilleures sources.

Avant de comparer les courbes obtenues, il est important de garder à l'esprit que chercher à obtenir par simulation les observations exactes réalisées par Seeley peut conduire à construire un modèle dédié à l'expérience tout en éloignant ce modèle du comportement d'une colonie. L'approche que nous proposons dans cet article consiste à modéliser les paramètres biologiques de la façon la plus fidèle possible afin de rendre compte de l'impact de chacun d'eux.

Il y a plusieurs différences entre nos résultats et les résultats réels, obtenus par Seeley. Premièrement, dans les premières 30 minutes (8:30:0), les Agents Butineuses retournent tous au moins une fois butiner à la source pour laquelle ils ont été entraînés, ce qui ne semble pas être le cas dans la réalité. La cause de cette différence n'est pas claire, mais pourrait par exemple provenir de la fraîcheur de la température, en cette heure matinale, qui freinerait certaines abeilles.

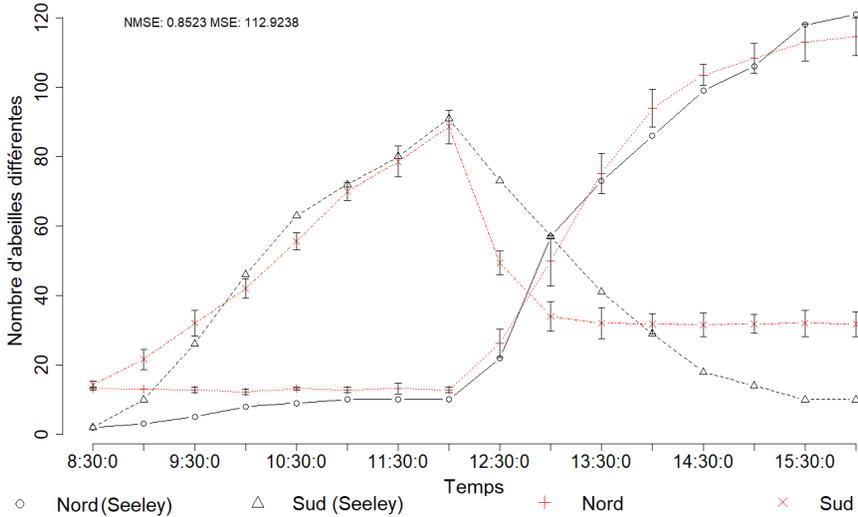


FIGURE 5.2. Le graphe montre l'évolution du nombre d'abeilles / d'Agents Butineuses différents sur chaque source, par tranche de 30 minutes. Superposées aux courbes obtenues par Seeley et al. [26], les courbes Nord (+) et Sud (x) montrent la moyenne obtenue sur 5 simulations, avec l'écart type représenté, à partir de la graine 0 du générateur aléatoire principal.

Une seconde différence réside dans la rapidité de l'abandon de la source Sud par les Agents, après 12:00:0, une fois la permutation effectuée. La concentration de la source Sud s'étant très appauvrie (0.75 mol/L contre 2.5 mol/L auparavant), la majorité des Agents revenant de butiner à cette source se basent sur sa nouvelle profitabilité pour prendre leur décision, et abandonner la source. D'après les résultats obtenus par Seeley, néanmoins, l'abandon semble se faire de manière beaucoup plus progressive. Là encore, plusieurs explications sont possibles : par exemple, il est possible que certaines butineuses, dans la réalité, n'abandonnent pas tout de suite une source suite à un butinage peu profitable, mais plutôt après plusieurs butinages, ce qui leur donnerait plus de flexibilité.

Enfin, il semblerait qu'environ 40 % des Agents recrutés sur la source Sud ne l'abandonnent pas (de 13:30:0 à 16:00:0, tout en ne recrutant pas d'autres Agents, alors que les résultats réels montrent que les abeilles ne sont que 20 % à continuer à y butiner. D'après nos recherches, avec les profils de la graine 0 du générateur aléatoire principal (dont une partie est représentée Figure 4.1) et l'état des stocks égal à 0.2, 71 Agents Butineuses sur les 200 continueraient de butiner à une source dont la concentration en sucre est égale à 0.75 mol/L. Une hypothèse permettant d'expliquer cette différence avec les données réelles est que, dans la colonie utilisée par Seeley, les profils des abeilles sont (génétiquement) plus sélectifs. Ce point de variabilité interindividuelle sera discuté ci-après.

5.2.2. Influence des paramètres

En plus de ces simulations, nous avons étudié l’influence des paramètres principaux, dans les intervalles décrits dans la Table 5.1. Les valeurs par défaut sont les valeurs obtenues par calibration, qui donnent dans la majorité des simulations des résultats proches des observations biologiques (Figure 5.2), en terme d’écart absolu ($MSE = 113$) et de forme de courbe ($NMSE = 0.85$).

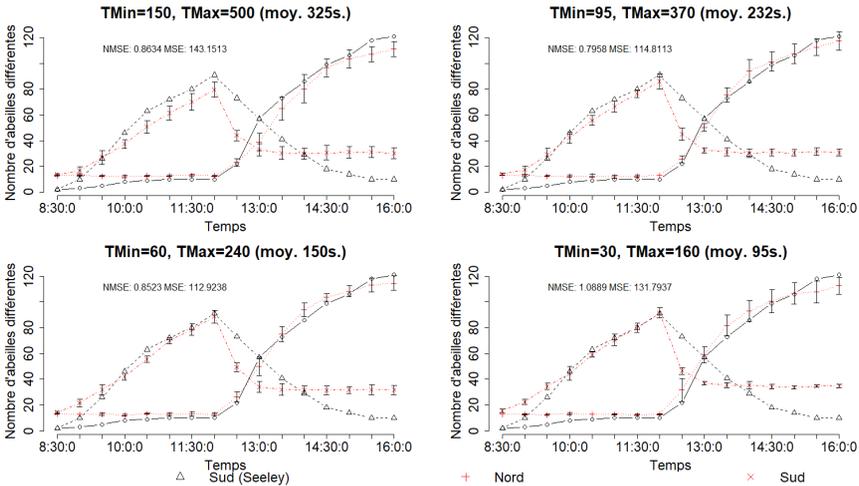


FIGURE 5.3. Même type de graphe que la Figure 5.2 avec la moyenne des courbes sur 5 essais pour différentes valeurs des paramètres T_{Min} et T_{Max} .

Il est intéressant de se pencher ici sur l’influence des paramètres T_{Min} et T_{Max} simulant le temps passé par les agents « sans-emploi » à chercher des danses. Concrètement, un agent « sans-emploi » attend un nombre de secondes tiré aléatoirement entre T_{Min} et T_{Max} avant de rechercher par un envoi de messages les agents en train de danser. La Figure 5.3 montre la moyenne des courbes obtenues avec la même graine 0 du générateur aléatoire principal sur 5 simulations, pour 4 valeurs différentes de T_{Min} et T_{Max} .

Sur chacune des courbes, ces paramètres influencent le taux de recrutement des agents : plus le temps passé à chercher des danses est grand, plus le nombre d’agents recrutés est faible. C’est particulièrement visible entre 8 : 30 : 0 et 10 : 00 : 0 pour la source Sud, et entre 12 : 00 : 0 et 13 : 00 : 0 pour la source Nord. Lorsque la piste de danse sera modélisée, nous pourrions mesurer le temps effectif passé par les agents à chercher des danses, et vérifier si l’on retrouve cette corrélation.

5.2.3. Importance de la variabilité interindividuelle

Nous avons également évalué l’impact de la variabilité interindividuelle dans le modèle. Les colonies d’*Apis mellifera* sont hétérogènes : toutes les abeilles ont

la même mère, la reine, mais peuvent être de pères différents. Cela signifie donc qu'il y a une certaine variabilité entre les individus, qui se traduit notamment pas des profils de fonction de stimulus-réponses différents. Il a été montré que cette diversité jouait un rôle important dans les systèmes complexes, par exemple dans la thermorégulation de la ruche [16].

Pour évaluer cet impact, nous avons comparé les résultats obtenus ci-dessus avec une colonie hétérogène, avec les résultats de la simulation de trois colonies homogènes, dans lesquelles les abeilles présentent des profils de fonction de stimulus-réponses identiques :

- P1 - Un profil avec une fonction qui possède un coefficient directeur proche de zéro, et qui amène les agents à ne danser que quelques secondes quelle que soit la profitabilité de la source : $0.11x - 1.5$;
- P2 - Un profil avec une fonction qui possède un coefficient directeur élevé, et qui rend les agents très discriminants dans leurs danses : $1.3x - 17.7$;
- P3 - Un profil avec une fonction qui possède un coefficient directeur intermédiaire : $0.5x - 7$.

Chacun de ces profils a été choisi pour permettre l'étape initiale de sélection et d'entraînement de quelques Agents sur les sources Nord et Sud : ces profils (avec un état des stocks à 0.2) les poussent à poursuivre le butinage sans recruter.

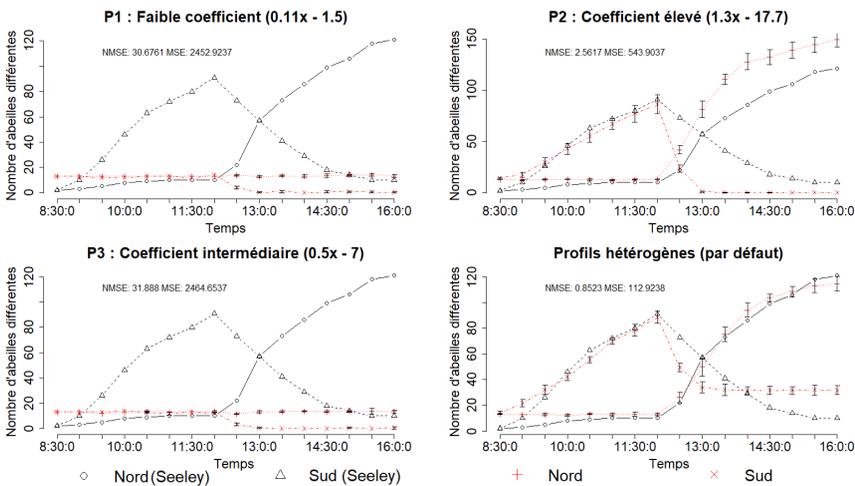


FIGURE 5.4. Même type de graphe que les Figures précédentes, avec la moyenne des courbes sur 5 essais pour chaque profil P1, P2 et P3.

Les courbes obtenues avec ces différents profils sont présentées dans la Figure 5.4. On peut y voir notamment que les abeilles les plus sélectives (P2) abandonnent en masse et très rapidement la source devenue moins profitable, alors que

les deux autres (*P1 et P3*) peinent à recruter, au point de quasiment abandonner les deux sources.

Concernant le profil *P2*, les résultats s'accordent avec ceux trouvés par Thenius et al. [30] : par rapport à une colonie hétérogène, les agents abandonnent de façon significative la source Sud plus rapidement, et recrutent également plus rapidement sur la source Nord (12:00:0 - 13:00:0). Cependant, le recrutement est moins efficace, dans le sens où le nombre de voyages effectués par les abeilles ayant déjà trouvé les sources est significativement plus bas. Ces agents, très sélectifs, vont en effet danser plus longtemps pour une source avec une forte profitabilité, et donc perdre en efficacité : le temps passé à recruter n'est pas passé à butiner.

En revanche, pour les profils *P1 et P3*, la faible durée (voire l'absence) des danses conduit à une absence de recrutement. On peut donc supposer que, dans une colonie hétérogène, les profils *P1, P2 et P3* sont complémentaires : les agents avec un profil *P1* se chargent du recrutement, et peuvent notamment recruter des agents avec les profils *P2, ou P3*, qui vont moins danser, et moins longtemps, mais être plus efficaces pour ramener des ressources à la ruche. Cela signifierait que les butineuses se détachent, *via* leur profil, en deux groupes : les danseuses, dont le rôle est de recruter le plus possible, et les transporteuses, qui vont moins recruter, et avoir donc plus de temps pour butiner. Ces résultats sont malgré tout préliminaires, et une analyse plus approfondie, ainsi que d'autres expériences, feront l'objet d'un futur article.

6. CONCLUSION ET PERSPECTIVES

Nous avons présenté dans cet article un modèle à base d'agents de butinage, et un Module Écosystème, qui permet aux agents de naviguer entre l'intérieur et l'extérieur de la ruche, environnements d'échelles et de granularité différentes.

Les résultats d'une première expérimentation vérifient tout d'abord l'influence de l'environnement de la ruche (météo, sources) sur les agents, et les mécanismes de recrutement implantés. Une seconde expérimentation, qui a visé à reproduire les résultats d'une expérimentation *in-vivo* faite par Seeley et al. [26], a donné des résultats qui valide ces mécanismes et le comportement des Agents Butineuses, ceux-ci étant suffisants pour mener à un processus d'auto-organisation de sélection des meilleures sources de nectar disponibles. L'influence des paramètres utilisés pour cette seconde expérimentation a fait l'objet d'une première analyse, qu'une analyse de sensibilité plus formelle pourra venir compléter. Enfin, les premiers résultats d'une étude de l'importance de l'hétérogénéité des fonctions de stimulus-réponses dans l'auto-organisation rejoignent ceux trouvés par Thenius et al. [30], et donneront lieu à de futurs travaux.

La mise en place du modèle à base d'agents de la colonie dans son ensemble, en prenant en compte plus de phénomènes complexes régissant la vie d'une colonie, est en cours et inclut une représentation spatiale explicite de la ruche et des stocks de ressources, ainsi que la définition des comportements des agents d'autres castes

et spécialités (e.g. la reine, les nourrices, les larves), et des interactions pouvant survenir entre eux et avec les Agents Butineuses. Dans la lignée de ce travail, nous présentons dans [1] un modèle complémentaire d'allocation et d'interruption de tâches qui permet de représenter l'auto-organisation des abeilles dans leurs différentes spécialités.

À terme, de nouvelles expériences conduites avec le modèle complet et le Module Écosystème sont prévues pour compléter la validation du modèle et comparer les résultats avec des données réelles sur la dynamique des colonies sur une année, par exemple. Enfin, il s'agira d'intégrer des moyens d'interaction et de visualisation avec la ruche, réalistes, intuitifs et ergonomiques [2].

BIBLIOGRAPHIE

- [1] T. ALVES, J. RIVIÈRE, C. ALAUX, Y. LE CONTE, F. SINGHOFF, T. DUVAL & V. RODIN, « An Interruptible Task Allocation Model: Application to a Honey Bee Colony Simulation », in *18th International Conference on Practical Applications of Agents and Multi-Agent Systems (PAAMS 2020)* (L'aquila, Italy) (Y. Demazeau, T. Holvoet, J. M. Corchado & S. Costantini, édés.), Lecture Notes in Artificial Intelligence, Springer, 2020.
- [2] T. ALVES, J. RIVIÈRE, V. RODIN & T. DUVAL, « Immersive and interactive visualisation of a virtual honey bee colony », EuroVR 2019, 2019, Poster.
- [3] M. A. BECHER, V. GRIMM, P. THORKEB et al., « BEEHAVE: a systems model of honeybee colony dynamics and foraging to explore multifactorial causes of colony failure », *Journal of Applied Ecology* **51** (2014), n° 2, p. 470-482.
- [4] F. BELLIFEMINE, F. BERGENTI, G. CAIRE & A. POGGI, « Jade — A Java Agent Development Framework », in *Multi-Agent Programming: Languages, Platforms and Applications* (R. H. Bordini, M. Dastani et al., édés.), Springer US, 2005, p. 125-147.
- [5] M. BETTI, J. F. LECLAIR, L. M. WAHL & M. ZAMIR, « Bee++: An Object-Oriented, Agent-Based Simulator for Honey Bee Colonies », *Insects* **8** (2017), n° 1, article no. 31.
- [6] E. BONABEAU, « Agent-based modeling: Methods and techniques for simulating human systems », *Proceedings of the National Academy of Sciences* **99** (2002), n° 3, p. 7280-7287.
- [7] E. BONABEAU, G. THERAULAZ, J.-L. DENEUBOURG et al., « Self-organization in social insects », *Trends in Ecology and Evolution* **12** (1997), n° 5, p. 188-193.
- [8] C. BOURJOT & V. CHEVRIER, « Multi-agent simulation in biology: application to social spiders case », in *Agent Based Simulation*, 2001, p. 18-23.
- [9] L. BRAUBACH, A. POKAHR, W. LAMERSDORF et al., « A generic simulation service for distributed multi-agent systems », in *From Agent Theory to Agent Implementation (AT2AI'04)*, 2004, p. 576-581.
- [10] G. CICH, S. GALLAND, L. KNAPEL et al., « Addressing the Challenges of Conservative Event Synchronization for the SARL Agent-Programming Language », in *Advances in Practical Applications of Cyber-Physical Multi-Agent Systems: The PAAMS Collection* (Y. Demazeau, P. Davidsson et al., édés.), Springer, 2017, p. 31-42.
- [11] A. DORNHAUS, F. KLÜGL, C. OECHSLEIN et al., « Benefits of recruitment in honey bees: effects of ecology and colony size in an individual-based model », *Behavioral Ecology* **17** (2006), n° 3, p. 336-344.
- [12] A. DROGOUL, « De la simulation multi-agents a la resolution collective de problèmes : une étude de l'émergence de structures d'organisation dans les systèmes multi-agents », Thèse, Paris 6, 1993, Dirigée par J. Ferber, J. Sciences appliquées, 382 pages.
- [13] A. DROGOUL, F. MICHEL & J. FERBER, « Multi-Agent Systems and Simulation: A Survey from the Agent Community's Perspective », in *Multi-Agent Systems – Simulation and Applications* (A. M. Uhrmacher & D. Weyns, édés.), Computational Analysis, Synthesis, and Design of Dynamic Systems, CRC Press / Taylor and Francis, 2009, p. 3-51.
- [14] Y. E. FIANYO, « Couplage de modèles à l'aide d'agents : le système OSIRIS », Thèse, Université Paris IX-Dauphine, 2001.

- [15] FIPA-ACL, « Speech acts library », Web, 2006.
- [16] J. C. JONES, M. R. MYERSCOUGH, S. GRAHAM & B. P. OLDROYD, « Honey Bee Nest Thermoregulation: Diversity Promotes Stability », *Science* **305** (2004), n° 5682, p. 402-404.
- [17] Y. KUBERA, P. MATHIEU & S. PICAULT, « How To Avoid Biases In Reactive Simulations », in *7th International Conference on Practical Applications of Agents and Multi-Agent Systems (PAAMS 2009)* (Y. Demazeau, J. Pavón et al., eds.), Springer Berlin Heidelberg, 2009, p. 100-109.
- [18] Y. KUBERA, P. MATHIEU & S. PICAULT, « IODA: an interaction-oriented approach for multi-agent based simulations », *Autonomous Agents and Multi-Agent Systems* **23** (2011), n° 3, p. 303-343.
- [19] M. MITCHELL, *Complexity: A Guided Tour*, Oxford University Press, 2009.
- [20] S. RUSSELL, A. B. BARRON & D. HARRIS, « Dynamic modelling of honey bee (*Apis mellifera*) colony growth and failure », *Ecological Modelling* **265** (2013), p. 158-169.
- [21] T. SCHMICKL & K. CRAILSHEIM, « HoPoMo: A model of honeybee intracolony population dynamics and resource management », *Ecological Modelling* **204** (2007), n° 1, p. 219-245.
- [22] T. SCHMICKL, R. THENIUS & K. CRAILSHEIM, « Swarm-intelligent foraging in honeybees: benefits and costs of task-partitioning and environmental fluctuations », *Neural Computing and Applications* **21** (2012), n° 2, p. 251-268.
- [23] T. D. SEELEY, « Division of labor between scouts and recruits in honeybee foraging », *Behavioral Ecology and Sociobiology* **12** (1983), n° 3, p. 253-259.
- [24] ———, « Honey bee foragers as sensory units of their colonies », *Behavioral Ecology and Sociobiology* **34** (1994), n° 1, p. 51-62.
- [25] ———, *The Wisdom of the Hive*, Harvard University Press, 1995.
- [26] T. D. SEELEY, S. CAMAZINE & J. SNEYD, « Collective decision-making in honey bees: how colonies choose among nectar sources », *Behavioral Ecology and Sociobiology* **28** (1991), n° 4, p. 277-290.
- [27] T. D. SEELEY & P. K. VISSCHER, « Assessing the benefits of cooperation in honeybee foraging: search costs, forage quality, and competitive ability », *Behavioral Ecology and Sociobiology* **22** (1988), n° 4, p. 229-237.
- [28] D. J. T. SUMPTER & D. S. BROOMHEAD, « Formalising the Link between Worker and Society in Honey Bee Colonies », in *Multi-Agent Systems and Agent-Based Simulation* (Berlin, Heidelberg) (J. S. Sichman, R. Conte & N. Gilbert, eds.), Springer Berlin Heidelberg, 1998, p. 95-110.
- [29] ———, « Shape and Dynamics of Thermoregulating Honey Bee Clusters », *Journal of Theoretical Biology* **204** (2000), n° 1, p. 1-14.
- [30] R. THENIUS, T. SCHMICKL & K. CRAILSHEIM, « The “Dance or Work” Problem: Why Do not all Honeybees Dance with Maximum Intensity », in *Multi-Agent Systems and Applications IV* (Berlin, Heidelberg) (M. Pěchouček, P. Petta & L. Z. Varga, eds.), Springer Berlin Heidelberg, 2005, p. 246-255.
- [31] K. S. TRAYNOR, Y. LE CONTE & R. E. PAGE, « Queen and young larval pheromones impact nursing and reproductive physiology of honey bee (*Apis mellifera*) workers », *Behavioral Ecology and Sociobiology* **68** (2014), n° 12, p. 2059-2073.
- [32] K. G. TROITZSCH, « Multi-Agent Systems and Simulation: A Survey from an Application Perspective », in *Multi-Agent Systems – Simulation and Applications* (A. M. Uhrmacher & D. Weyns, eds.), Computational Analysis, Synthesis, and Design of Dynamic Systems, CRC Press / Taylor and Francis, 2009, p. 53-75.
- [33] H. DE VRIES & J. C. BIESMEIJER, « Modelling collective foraging by means of individual behaviour rules in honey-bees », *Behavioral Ecology and Sociobiology* **44** (1998), n° 2, p. 109-124.
- [34] U. WILENSKY, « NetLogo », <http://ccl.northwestern.edu/netlogo/>, Center for Connected Learning and Computer-Based Modeling, 1999.
- [35] M. L. WINSTON, *The Biology of the Honey Bee*, Harvard University Press, 1987.

ANNEXES

TABLE 6.1. Les probabilités et principaux paramètres utilisés par l'Agent Eco-système et l'Agent Butineuse. Chaque valeur est donnée par rapport au ratio $\frac{\text{temps simulé}}{\text{temps réel}} = \frac{1}{1000}$. Les probabilités pondérées par le stock des ressources de la colonie sont représentées avec une *.

Probabilité/Paramètre	Valeur	Adapté de
Agent Ecosystème:		
p_{mort}	0.0108	[20]
$p_{succès}$:		
Pour l'Agent Butineuse recruté		
- trouve la source au 1 ^{er} essai	0.2	[25]
- trouve la source au 2 ^e essai	0.4	[25]
- trouve la source au 3 ^e essai	0.6	[25]
- trouve la source au 4 ^e essai	0.8	[25]
- au-delà, et Agent entraîné	0.93	[27]
Pour l'Agent Butineuse « scout »	0.43	[27]
Durée du butinage pour une source à 650 m	[5.41; 11.37] min.	[26]
Quantité max. en un seul voyage :		
- de nectar	[25; 60] mg	[11]
- de pollen	[14.6; 16] mg	[11]
Vitesse de vol	0.774 m.s ⁻¹	[26]
Consommation de nectar en vol	0.003 mg.m ⁻¹	[25]
Agent Butineuse :		
p_{scout} *	0.00825	[11]
p_{piste_danse} *	0.00005	Déduit de la calibration
Nombre max. de voyages d'affilé		
	[20; 30]	[25]
Température de sortie		
- minimum	[10; 12]°C	[25]
- maximum	[39; 41]°C	[25]
Temps d'attente entre deux recherches de danse		
- T_{Min}	60 s	Déduit de la calibration
- T_{Max}	240 s	Déduit de la calibration
Nombre de <i>waggle</i> à suivre pour comprendre une danse		
- W_{Min}	5	[25]
- W_{Max}	15	[25]

ABSTRACT. — The agent-based approach has been successfully used in the past years to model and simulate complex systems. We use this approach on a honeybee colony in a Dadant hive, where several tens of thousands of bees interact, in order to evaluate the impact of local actions at the bee-level (such as beekeeping practices) on the global system. In this article, we focus on the foraging activity, its self-organisation mechanisms and the behaviour of foraging bees, and how these bees interact with the environment of the hive, greatly different in granularity and scale. We present a customizable, agent-compliant module that aims at modelling and simulating the foraging, according to the weather and the surrounding nectar sources. The results of two experimentations provide a first validation of our model, showing that the agents' behaviours lead to a self-organizing process of the best available sources' selection.

KEYWORDS. — Multiagent simulation, Self-organization, Environment, Complex Systems.

Manuscrit reçu le 16 juillet 2021, révisé le 31 janvier 2022, accepté le 15 mars 2022.