



Modélisation comportementale selon différentes échelles temporelles pour des trajectoires issues de scènes encombrées

Guillaume Chiron, Petra Gomez-Krämer, Michel Ménard

► To cite this version:

Guillaume Chiron, Petra Gomez-Krämer, Michel Ménard. Modélisation comportementale selon différentes échelles temporelles pour des trajectoires issues de scènes encombrées : Approche bayésienne non paramétrique par HDPs. *Revue des Sciences et Technologies de l'Information - Série RIA : Revue d'Intelligence Artificielle*, 2015, Réseaux bayésiens et modèles probabilistes, 29 (2), pp.173-203. hal-01243691

HAL Id: hal-01243691

<https://hal.science/hal-01243691>

Submitted on 24 Feb 2016

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.

Modélisation comportementale selon différentes échelles temporelles pour des trajectoires issues de scènes encombrées

Approche Bayésienne non-paramétrique par HDPs

Guillaume Chiron, Petra Gomez-Krämer, Michel Ménard

Université de La Rochelle – Laboratoire L3i

Avenue Michel Crépeau

17042 La Rochelle CEDEX 1, France

{guillaume.chiron,petra.gomez,michel.menard}@univ-lr.fr

RÉSUMÉ. Cet article s'intéresse au problème de la modélisation comportementale à partir des trajectoires issues de scènes encombrées, en particulier lorsque la connaissance métier sur les données est limitée. Nous proposons d'utiliser une approche Bayésienne non-paramétrique – le MLC-HDP défini par D. Wulsin – pour découvrir et modéliser des comportements récurrents à différentes échelles temporelles. Ainsi, l'analyse exploratoire des trajectoires s'effectue par classification non-supervisée, simultanément à des niveaux sémantiques différents et où le nombre de clusters pour chaque niveau n'est pas défini a priori mais est estimé à partir des données. Dans un premier temps, notre approche est validée à l'aide d'une pseudo-vérité terrain générée à partir d'un système multi-agents. Ce dernier permet de simuler, avec relativement peu d'a priori, des comportements distincts et des trajectoires associées. Dans un deuxième temps, nous testons notre approche sur des données de trajectoires réelles d'abeilles.

ABSTRACT. This article addresses the problem of behavioral modeling based on trajectories extracted from crowded scenes, especially in the situation of limited knowledge on the data. We propose to use a nonparametric Bayesian approach – the MLC-HDP defined by D. Wulsin – to discover and model reoccurring behaviors at different time scales. Thus, the exploratory analysis of trajectories is performed by an unsupervised classification, simultaneously over different semantic levels, with the number of clusters for each level not defined a priori but estimated from the data. Firstly, we validated our approach using a pseudo-ground truth generated using a multi-agent system which is able to simulate, with relatively low a priori, distinct behaviors and associated trajectories. Secondly, we tested our approach on real honeybee trajectories.

MOTS-CLÉS : Modélisation comportementale, Approche Bayésienne non-paramétrique, Processus hiérarchique de Dirichlet, Colonie d'abeilles, Surveillance apicole.

KEYWORDS: Behavioral modeling, Bayesian nonparametric approach, Hierarchical Dirichlet process, Beehive monitoring, Honeybee colony.

RIA. 1-31 - Version auteur incluant les corrections demandées par les pairs

1. Introduction

Contexte. Les communautés "vision par ordinateur" et "intelligence artificielle" s'intéressent depuis plusieurs années aux problèmes de la modélisation comportementale à partir de séquences vidéo. Les applications qui en découlent sont nombreuses. On peut citer notamment l'automatisation des traitements pour la vidéosurveillance (e.g. recherche de contenus similaires ou anormaux). Les travaux présentés dans cet article s'inscrivent dans le cadre d'un projet interdisciplinaire bioinformatique visant à étudier le comportement d'une colonie d'abeilles à l'aide de séquences vidéo acquises devant la ruche. Le caractère innovant et exploratoire de ce type d'application provient également de l'absence de connaissances métiers (en termes de dynamiques de mouvement et de comportement). Nous sommes ici aussi contraints par la nature erratique des mouvements (i.e. abeilles en vol rapide dans un volume 3D). Un calcul du mouvement dense via des approches de type flot optique n'est pas envisageable pour ce type de vidéo dit "encombrée" que nous souhaitons traiter. C'est pourquoi nous proposons d'utiliser les trajectoires 3D de chaque individu en déplacement dans la scène. La méthode de collecte des trajectoires des abeilles en vol est détaillée par Chiron *et al.* (2013).

Problématique. Comment, avec un minimum d'*a priori*, peut-on détecter et modéliser des comportements récurrents à partir d'un ensemble de trajectoires erratiques ? Au vu de la complexité des données à modéliser et des hypothèses très limitées qu'il nous fût possible d'élaborer, nous avons choisi d'aborder le problème avec des outils qui nécessitent peu d'*a priori* dans leur mise en œuvre, à savoir les approches Bayésiennes non-paramétriques, et particulièrement leurs extensions hiérarchiques tel que le HDP (pour Hierarchical Dirichlet Process) proposé par Teh *et al.* (2006). Celles-ci ont l'avantage de pouvoir combler le manque d'*a priori* disponible sur les données par la construction d'une structure hiérarchique.

Littérature. Depuis quelques années, malgré un contexte formel relativement complexe et un temps de calcul qui peut s'avérer important, les modèles Bayésiens non-paramétriques connaissent un succès grandissant dans plusieurs domaines tels que la génétique (cf. Beal, Krishnamurthy, 2006), la finance (cf. Griffin, Steel, 2006), et bien sûr, la vision par ordinateur. Pour ne citer que quelques exemples, on peut mentionner des applications liées à l'analyse de mouvements (cf. Fox *et al.*, 2007), à la catégorisation de scènes (cf. Fei-Fei, Perona, 2005), à la reconnaissance d'objets (cf. Sudderth *et al.*, 2005) ou à la reconnaissance d'activités humaines (cf. Niebles *et al.*, 2008).

Approche proposée. Nous mettons en œuvre une extension particulière du HDP, nommée MLC-HDP (pour multi-level clustering hierarchical Dirichlet Process) qui fut originalement proposé par Wulsin *et al.* (2012) dans le cadre d'une application médicale. Cette méthode permet de regrouper des données simultanément selon plusieurs niveaux obéissant à une structure hiérarchique. Dans ce contexte, nous proposons de modéliser le comportement d'un ensemble d'individus (e.g. colonie d'abeilles) à partir d'une classification non-supervisée de sous-séquences, appliquée simultanément à des échelles temporelles différentes. Ces sous-séquences sont issues d'un découpage

arbitraire du corpus vidéo. Le découpage ne s'appuie donc pas sur une méthode de détection de changement de plan, mais simplement sur une fenêtre d'une durée fixée. Ainsi, le corpus est exprimé sous la forme d'une structure de données hiérarchique à différents grains (e.g. entités temporelles relativement longues appelées "*captures*" d'environ 15 min, constituées d'entités temporelles plus courtes appelées "*sousclips*" d'environ 1 min). Ces entités temporelles englobent l'information de mouvement de manière éparse à l'aide d'un ensemble de trajectoires. Nous verrons ainsi comment cette structure multi-échelle permet au travers du MLC-HDP de découvrir et de modéliser des concepts sémantiques (i.e. "*activités*" et "*comportements*") à partir de données de trajectoires.

Validation par simulation. Les biologistes s'intéressent de plus en plus aux problèmes de la modélisation de l'activité de colonies à partir de vidéos (e.g. travaux de Schaefer, Claridge-Chang (2012)). Mais à ce jour, il n'existe pas à notre connaissance de vérité terrain faisant le lien entre trajectoires et comportements d'une colonie d'insectes. Pour palier ce manque, en plus des expérimentations sur des données réelles, nous proposons une méthode originale pour valider notre approche de modélisation comportementale. Celle-ci repose sur l'utilisation d'un Systèmes Multi-Agents (SMA) d'abeilles capable à la fois de faire émerger des comportements et de générer des trajectoires "non contraintes" associées à chacun des individus.

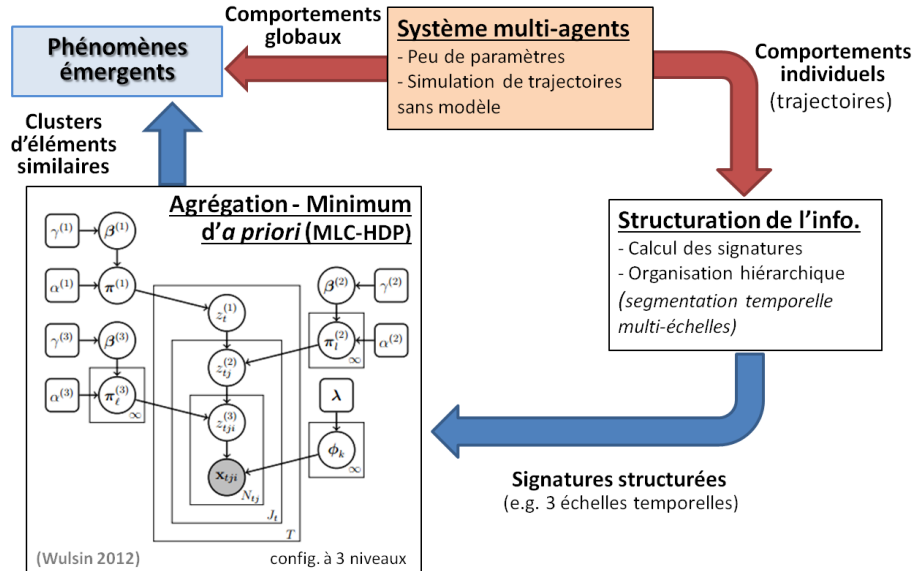


Figure 1. Schéma global de notre chaîne de montée en sémantique. La partie supérieure (SMA) illustre le côté "simulation", tandis que les parties inférieures (Structuration + Agrégation) illustrent le côté "modélisation".

Ainsi, la figure 1 illustre la chaîne de montée en sémantique que nous proposons de détailler dans cet article. Cette chaîne comprend à la fois des aspects "simulation"

de données brutes par SMA, et des aspects "modélisation" de phénomènes émergents de plus hauts niveaux.

Plan. La section 2 présente les approches existantes de modélisation comportementales et aborde également les notions nécessaires à la compréhension des approches Bayésiennes non-paramétriques, et en particulier celles à base de processus de Dirichlet. La section 3 expose notre solution s'appuyant sur le MLC-HDP et une structure hiérarchique pour la modélisation de comportements à partir d'un ensemble de trajectoires. Puis, la section 4 concerne la validation de l'approche proposée à l'aide d'une simulation multi-agents. Enfin, la section 5 montre des résultats sur des données de trajectoires issues de vidéos non simulées d'abeilles.

2. État de l'art sur les modèles Bayésiennes non-paramétriques et la modélisation comportementale

2.1. Positionnement dans le domaine de la modélisation comportementale

Dans la littérature, on trouve un large panel de travaux traitant du problème de la modélisation de comportements à partir de vidéos. Le lecteur peut se référer à l'étude de Morris, Trivedi (2008) qui récapitule un bon nombre d'approches existantes. On distingue les deux catégories suivantes :

Approches s'intéressant aux trajectoires individuelles. Celles-ci exploitent des données de trajectoires obtenues à partir du suivi de cibles. Les trajectoires représentent de manière éparse l'essentiel de la dynamique de la vidéo. Les approches, s'appuyant sur les trajectoires, s'intéressent le plus souvent aux signatures invariantes et se focalisent principalement sur l'information spatiale locale. Pour ne citer que quelques exemples, la méthode de Stauffer, Grimson (2000) regroupe les trajectoires en activités à partir, entre autres, des positions, vitesses et directions. Wang *et al.* (2006) utilisent le même principe en s'appuyant sur une distance de Hausdorff modifiée pour comparer les trajectoires entre elles. On retrouve dans d'autres travaux tels que ceux de Hongeng, Nevatia (2001); Feldman, Balch (2004) des méthodes de découpage de trajectoires en primitives. Celles-ci sont souvent obtenues à l'aide d'un classifieur entraîné sur une base de données étiquetées. On trouve aussi des travaux qui proposent de modéliser les interactions entre les cibles, par exemple avec un Modèle de Markov Caché (ou HMM pour Hidden Markov Model) comme Oliver *et al.* (2000) ou des champs aléatoires de Markov (ou MRF pour Markov Random Field) comme Khan *et al.* (2005). Quant à eux, Intille, Bobick (1999) utilisent un réseau Bayésien pour analyser les stratégies d'un jeu de football, mais le suivi de cibles automatisé n'ayant pas fonctionné, les trajectoires des joueurs ont été annotées à la main.

Approches s'appuyant sur l'estimation du mouvement. Celles-ci exploitent des données de mouvement telles que des vecteurs calculés par une méthode de flot optique. Ces vecteurs de mouvement représentent, pour chaque image de la vidéo, la dynamique de la scène dans son ensemble de manière dense. Ces approches sont plutôt utilisées sur des scènes encombrées pour lesquelles un suivi de cibles de ma-

nière individualisée n'est pas envisageable. Ces approches s'appuient sur l'évolution du mouvement et attribuent généralement plus d'importance à la dimension temporelle que spatiale pour la modélisation des comportements. Se pose alors le problème de savoir quand un comportement (ou une activité) commence et se termine. Quelques publications abordent ces problèmes, tels que (Zelnik-Manor, Irani, 2001) pour la modélisation et le clustering de clips vidéo à l'aide d'un histogramme multirésolution, ou (Zhong *et al.*, 2004) pour l'analyse de type sujets/documents sur de la vidéo. Mais il reste à résoudre le problème de la modélisation des activités se déroulant simultanément. En réponse, des travaux comme ceux de Rao *et al.* (2002); Gorelick *et al.* (2007) proposent de découper le concept d'activité en différentes actions.

Comme nous l'avons mentionné dans l'introduction, ces dernières années ont montré un intérêt grandissant pour les modèles Bayésiens non-paramétriques, notamment dans le domaine de la modélisation d'activités ou de comportements. Les modèles Bayésiens hiérarchiques fournissent d'ailleurs une solution attractive pour la modélisation à partir de données chaotiques. Le HDP originalement proposé par Teh *et al.* (2006), est une approche Bayésienne non-paramétrique permettant de classer de manière non-supervisée un jeu de données à différentes échelles, tout en partageant l'information au travers d'une hiérarchie. Des approches similaires ont été largement utilisées par la communauté document (e.g. LDA, pour Latent Dirichlet Allocation, dans le cadre de l'analyse de sujets/documents). Récemment, ces approches HDP ont montré leur efficacité pour regrouper des données de natures différentes. Concernant la modélisation d'activités à partir de vidéos, la majorité des travaux s'appuyant sur des modèles Bayésiens non-paramétriques, quelque soit la structure de données sous-jacente, utilise des données de mouvement issues du flot optique, où les vidéos sont scindées en entités temporelles (e.g. quelques secondes) (cf. Kuettel *et al.*, 2010; Wang *et al.*, 2009; Hospedales *et al.*, 2009). Il existe relativement peu de travaux reposant sur des modèles Bayésiens non-paramétriques à partir de données de trajectoires. Dans le domaine médical, on peut citer Wang *et al.* (2011) qui modélisent des tractographies (données proches des trajectoires) à l'aide d'un modèle de mélange de HDPs.

Dans la suite de l'état de l'art, nous nous intéresserons aux modèles Bayésiens non-paramétriques, et verrons notamment comment ces derniers peuvent être utilisés pour la modélisation de comportements à partir de trajectoires.

2.2. Présentation succincte du processus de Dirichlet et quelques extensions

Cette section présente succinctement différentes approches de clustering basées sur le processus de Dirichlet. En premier lieu, nous invitons le lecteur non familier avec ce type d'approche à lire la section Annexe qui pose les bases théoriques minimums. Pour plus de détails sur les fondements, on conseille au lecteur de se reporter aux tutoriaux de référence (cf. Teh, 2007; Teh, Jordan, 2010; Orbanz, Teh, 2011; Walker *et al.*, 1999). Dans cette section, nous abordons aussi les extensions du processus de Dirichlet telles que le MLC-HDP proposé par Wulsin *et al.* (2012) qui est l'un des maillons clés de notre approche.

2.2.1. Processus de Dirichlet (DP)

Le processus de Dirichlet (DP) introduit par Ferguson (1973) est une généralisation de la loi de Dirichlet pour K (nombre de cluster) infini. Un DP permet de tirer une distribution presque sûrement discrète (appelée G) à partir d'une distribution de probabilité continue (appelée H ou G_0). Le tirage de G par un DP est noté $G \sim DP(\alpha, H)$, avec α (parfois noté γ) un scalaire représentant la concentration (ou paramètre d'échelle). On note pour toutes partitions $A_1, \dots, A_K$ d'un espace mesurable :

$$(G(A_1), \dots, G(A_K)) \sim \text{Dir}(\alpha H(A_1), \dots, \alpha H(A_K)).$$

La figure 2 illustre la notion de partitionnement infini.

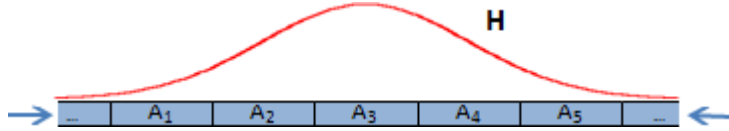


Figure 2. Exemple où H est une loi de probabilité normale (1 dimension). La densité est partitionnée, et lorsque $K \rightarrow \infty$, chaque A_i devient une impulsion de Dirac.

Les DP peuvent être construits de plusieurs manières : la représentation du "bâton cassé" (ou en anglais "stick-breaking") (cf. Sethuraman, 1994), la métaphore du restaurant chinois (cf. Teh *et al.*, 2006) ou également la méthode des urnes de polya (cf. Blackwell, MacQueen, 1973). Nous détaillons ci-dessous la construction stick-breaking.

2.2.2. Construction "stick-breaking"

L'objectif est d'associer des poids π (ou masses) tendant à être de plus en plus faibles à l'infini des impulsions de Dirac (appelées atomes) support de G . Rappelons que ces atomes sont tirés via la loi de Dirichlet (conjuguée de la loi multinomiale) définie par H . Le paramètre α contrôle la distribution des masses à travers les atomes. Plus α est grand, plus la masse est concentrée sur les premiers atomes. k désigne le cluster associé à un poids. La notation suivante accompagnée de la figure 3 illustre le principe.

$$\begin{aligned} \beta_i &\sim \text{Beta}(1, \alpha) \quad \text{avec } i = 1, \dots, \infty \\ \pi_i &= \beta_i \prod_{\ell=1}^{i-1} (1 - \beta_\ell) \\ \phi_i &\sim H \\ G &= \sum_{k=1}^{\infty} \pi_k \delta_{\phi_k} \end{aligned}$$

Cette mesure est parfois notée $\pi \sim GEM(\alpha)$ faisant référence à Griffiths, Engen, and McCloskey (cf. Pitman, 2002).

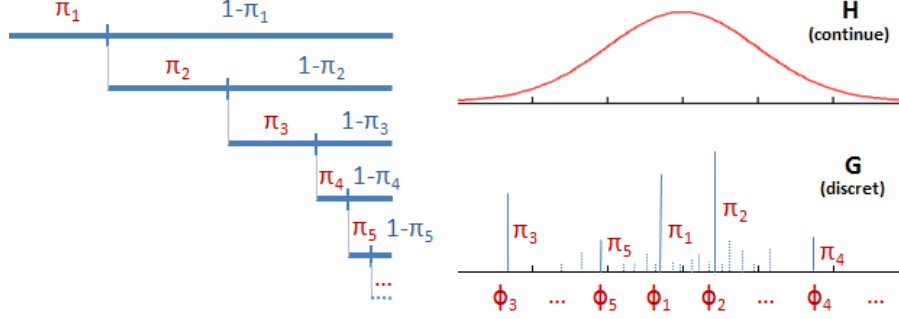


Figure 3. Construction stick-breaking du support de la distribution discrète G à partir de la distribution H .

2.3. Extensions du Processus de Dirichlet

2.3.1. Processus de Dirichlet à mélange (DPM)

Grâce aux Processus de Dirichlet à Mélange (DPM ou Mixture par Processus de Dirichlet), il devient possible de faire du clustering sur des données avec un nombre de clusters K théoriquement infini. L'aspect clustering vient du fait qu'à partir d'un DP, on peut tirer une distribution discrète G constituée d'atomes où chacun représente un cluster potentiel. Considérant un tirage à partir de G , les chances de retomber sur les mêmes clusters de poids élevé provoquent un effet clustering.

De manière formelle, on suppose qu'une observation x_i est tirée d'une distribution F (par exemple une Gaussienne) de paramètre θ_i , lui-même tiré de G , issue d'un DP. Ci-dessous la notation d'un DPM, accompagnée de la figure 4 pour sa représentation graphique. Notons que G modélise la distribution des clusters, θ_i désigne les paramètres du cluster associé l'observation x_i .

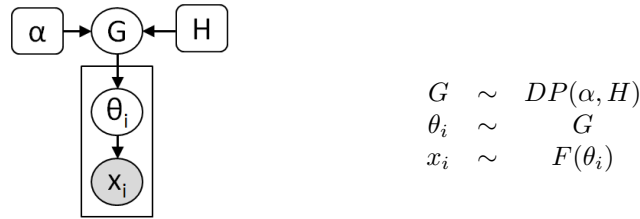


Figure 4. Représentation graphique et équations d'un DPM.

Notons que la loi *a posteriori* d'un DP est aussi un processus de Dirichlet, noté :

$$P(G|\theta_1, \dots, \theta_n) = DP\left(\frac{\alpha}{\alpha + n}H + \frac{1}{\alpha + n} \sum_{i=1}^n \delta_{\theta_i}, \alpha + n\right).$$

Cela permet notamment d'estimer H , la distribution de base sur l'espace des paramètres (e.g. μ et θ de lois Gaussiennes) servant à modéliser les clusters à partir des observations. Après plusieurs répétitions au sein d'un processus itératif, on conserve la solution qui offre la plus grande vraisemblance. Finalement, l'algorithme pour la mise en pratique est relativement simple (cf. Algorithme 1). On utilise pour estimer la densité de probabilité *a posteriori* des paramètres une approche MCMC et plus particulièrement un échantillonneur de type Gibbs.

```

while stopping criterion not satisfied do
  for each observation  $x_i$  do
    1) Remove  $x_i$  from its assigned cluster, and delete the cluster if empty.
    2) Re-estimate the parameters (e.g. Gaussian law) for each cluster.
    3) Compute the likelihood of  $x_i$  to be generated by each existing cluster
       into a probability vector. A value  $\alpha$  is added the vector corresponding
       the case of a new cluster.
    4) Sample the new class (i.e. assigned cluster) of  $x_i$  according to the
       distribution (i.e. multinomial) given by the DP.
    5) If last position is chosen, a new cluster sampled from  $G$  is activated.
       In practice, we can estimate the cluster parameters from  $x_i$ 
  end
end

```

Algorithme 1 : Méthode MCMC pour la mise en pratique du clustering par DP.

La figure 5 montre un exemple de clustering sur des données aléatoirement générées par un mélange de Gaussiennes. On note la capacité du clustering par DP à proposer des solutions avec des clusters se chevauchant (ou se superposant complètement), contrairement à une approche classique de type K-means.

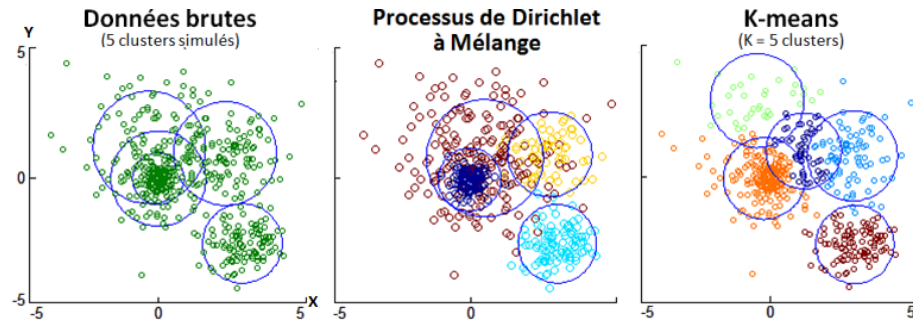


Figure 5. Comparaison de la méthode K-means et DP sur des données générées aléatoirement à partir d'un mélange de Gaussiennes. À gauche, pour la méthode K-means, K est calculé automatiquement via une méthode agglomérative à base de dendrogrammes. À droite, le résultat correspond à la 100ème itération d'un DP arrivé à un niveau relativement stable.

2.3.2. Processus Hiérarchique de Dirichlet (HDP)

Le Processus Hiérarchique de Dirichlet (ou HDP pour Hierarchical Dirichlet Process) proposé par Teh *et al.* (2006) est une manière de clusteriser des données potentiellement divisées en plusieurs groupes. La classification non-supervisée s'effectue au sein de chaque groupe indépendamment. Les clusters sont cependant partagés entre les groupes (cf. Figure 6). Tout comme un DP, le HDP permet de trouver le nombre de clusters et leurs paramètres susceptibles d'expliquer les données en s'appuyant sur les données elles-mêmes. Mais le principal intérêt du HDP réside dans sa capacité à partager l'ensemble des clusters au sein des différents groupes. À chaque cluster est associé un poids (possiblement nul) par groupe. Ainsi, les clusters peuvent n'être associés à aucune donnée dans certains groupes. Un HDP est défini par un DP classique, et plusieurs sous-DPs dépendants (1 associé à chaque groupe). Le DP classique offre le support sur lequel les sous-DPs vont s'appuyer pour échantillonner des observations. La figure 7 introduit la notation du HDP et sa représentation graphique correspondante. Notons que G_j modélise la distribution des clusters pour le groupe j , θ_{ji} désigne les paramètres du cluster associé à la i ème observation dans le groupe j . x_{ji} correspond donc à l'observation i générée à partir du cluster associé.

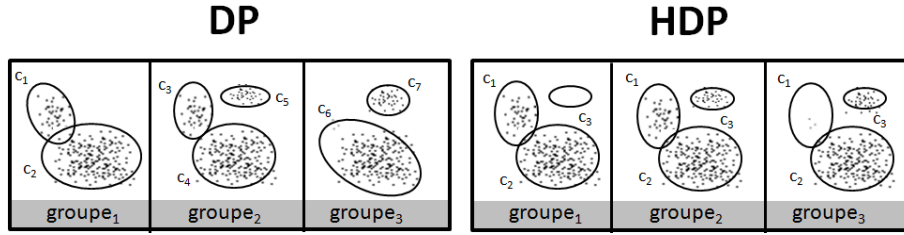


Figure 6. Illustration de la faculté de partage des clusters de la méthode HDP sur des données structurées en groupes, comparée à l'approche DP classique.

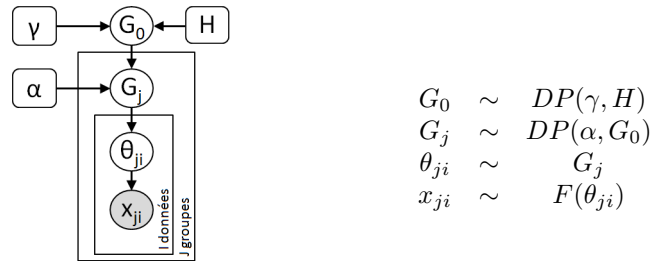


Figure 7. Représentation graphique et équations d'un HDP.

La nouvelle construction "stick-breaking" pour le HDP est donnée par :

$$\begin{aligned}
\beta_{0k} &\sim \text{Beta}(1, \gamma) \\
\pi_{0k} &= \beta_{0k} \prod_{l=1}^{k-1} (1 - \beta_{0l}) \\
\theta_k^* &\sim H \\
G_0 &= \sum_{k=1}^{\infty} \pi_{0k} \delta_{\theta_k^*} \\
\beta_{jk} &\sim \text{Beta}(\alpha \beta_{0k}, \alpha (1 - \sum_{l=1}^k \beta_{0l})) \\
\pi_{jk} &= \beta_{jk} \prod_{l=1}^{k-1} (1 - \beta_{jl}) \\
G_j &= \sum_{k=1}^{\infty} \pi_{jk} \delta_{\theta_k^*}
\end{aligned}$$

2.3.3. Processus hiérarchique de Dirichlet multi-niveaux (MLC-HDP)

Le clustering par processus hiérarchique de Dirichlet multi-niveaux introduit par Wulsin *et al.* (2012) combine les avantages du HDP et du NDP (Nested HDP proposé par Rodríguez *et al.* (2008)). Il intègre l'aspect clustering à plusieurs niveaux comme pour le NDP tout en gardant la capacité du HDP à fournir un support commun pour le partage des clusters au travers des données. La flexibilité du MLC-HDP permet de définir le nombre de niveaux en fonction de la structure de données utilisée. Voici par exemple la définition d'un MLC-HDP à deux niveaux et sa représentation graphique associée.

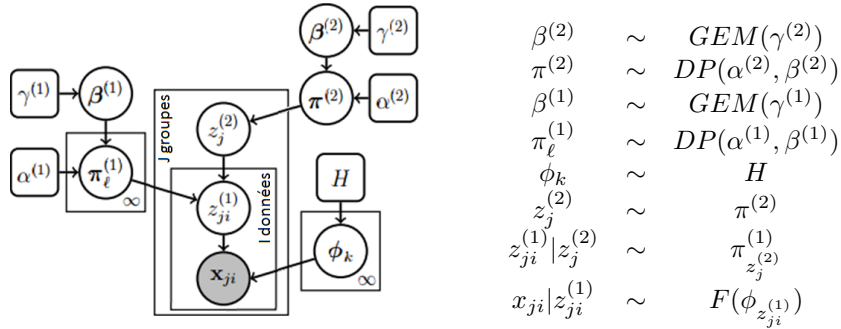


Figure 8. Représentation graphique et équations d'un MLC-HDP à deux niveaux.

Dans le modèle MLC-HDP défini ci-dessus, les indices supérieurs indiquent le niveau au sein de la structure hiérarchique. $\pi^{(n)}$ représente l'ensemble des poids associés à chaque cluster au niveau n . L'indice inférieur associé à π renseigne le cluster de niveau supérieur décrit. z renseigne le cluster d'appartenance d'une donnée (pour le niveau de base) ou d'une entité de plus haut niveau (lorsque le niveau > 1). Le modèle s'appuie sur un certain nombre d'hyper-paramètres (α et γ à chaque niveau), néanmoins, Wulsin *et al.* (2012) ont montré que l'*a priori* porté par ces hyper-paramètres a une moindre influence sur les résultats finaux.

3. Modèle MLC-HDP pour des trajectoires issues de séquences vidéo

Les approches Bayésiennes non-paramétriques de type LDA ou HDP ont récemment montré leur efficacité en termes de modélisation comportementale de scènes

encombrées. Nous proposons ainsi dans cette section d'utiliser le MLC-HDP de Wulsin *et al.* (2012) originalement proposé dans le cadre d'une application médicale pour modéliser des comportements à partir de données structurées de trajectoires. Le MLC-HDP a l'avantage de gérer des structures de données possédant plus de deux niveaux hiérarchiques, ce qui, à notre connaissance, va au delà des approches similaires existantes (cf. Kuettel *et al.*, 2010; Wang *et al.*, 2009; Hospedales *et al.*, 2009). De plus, plutôt que s'appuyer, comme il se fait classiquement pour les scènes encombrées, sur des données de mouvement dense (e.g. flot optique), nous proposons d'utiliser des données éparées, et en particulier des trajectoires individuelles.

3.1. Structure de données

Le jeu de données utilisé dans notre approche est structuré sous la forme d'un ensemble de "captures". Les captures correspondent à une même scène filmée par une caméra fixe à différents instants (e.g. heures, jours, semaines). Les captures sont segmentées en "sousclips" à l'aide d'une fenêtre glissante non-chevauchante (e.g. 1 min). La taille de la fenêtre doit être suffisamment grande pour assurer un nombre suffisant de trajectoires dans chaque sousclip, mais doit aussi être de taille raisonnable pour éviter que les sousclips contiennent plusieurs "activités". Le concept d'activité sera défini par la suite à l'aide d'exemples. Finalement, le jeu de données est constitué de nombreuses trajectoires structurées en captures et sousclips suivant une granularité donnée. La figure 9 illustre notre structure de données constituée de 3 niveaux.

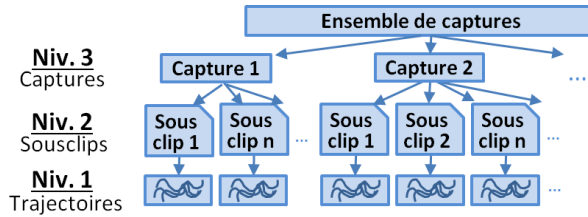


Figure 9. Structure de données hiérarchisée en captures, sousclips et trajectoires.

3.2. Clustering selon trois niveaux

Allant de pair avec la structure de données, nous utilisons une version à trois niveaux du MLC-HDP, où les éléments (captures, sousclips, trajectoires) issus d'un même niveau sont regroupés au sein de clusters types en fonction de leurs similarités. Voici comment le modèle à trois niveaux s'intègre avec les données.

Niveau 1 : les trajectoires, ou plutôt signatures de trajectoires, sont regroupées en clusters appelés "atomes d'activité". Par exemple, un cluster peut regrouper des cibles (ou plutôt des trajectoires) se dirigeant dans la même direction;

Niveau 2 : chaque sousclip est modélisé par une expression pondérée de tous les atomes d'activité. Les modèles d'expressions (ou ensembles de poids) d'atomes d'activité proches sont regroupés en modèles types que l'on appelle activités. Donc, à

chaque *sousclip* est associée une *activité*. Par exemple, une *activité* peut modéliser le fait que les individus d'une colonie effectuent des allers/retours entre un point A et un point B;

Niveau 3 : chaque *capture* est modélisée comme une expression pondérée de toutes les *activités*. Les modèles d'expressions (ou ensembles de poids) d'*activité* proches sont regroupées en modèles types que l'on appelle "*comportements*". Donc, à chaque *capture* est associée un *comportement*. Par exemple, un *comportement* peut modéliser le fait qu'une colonie d'individus passe 20% du temps à explorer l'espace, et 80% du temps à exploiter une ressource de manière efficace.

De manière formelle, nous considérons un jeu de données constitué de T *captures*, où chaque *capture* t est composée de J_t *sousclips*, où chaque *sousclip* j est composé de N_{tj} signatures de trajectoires et où chaque signature i s'écrit $x_{tji} \in \mathbb{R}^d$ avec d la dimension spatiale des signatures. L'observation $\{\{\{x_{tji}\}_{i=1}^{N_{tj}}\}_{j=1}^{J_t}\}_{t=1}^T$ de toutes les trajectoires est modélisée par un ensemble de processus hiérarchiques de Dirichlet. Sur le niveau le plus bas (niveau 1), les trajectoires sont regroupées au sein de clusters définis par des DP dont les paramètres sont tirés d'une distribution génératrice H . Les paramétrés γ et α sont les hyperparamètres du modèle. Nous considérons F comme une distribution normale multivariée avec une matrice de covariance diagonale. Les trois niveaux du MLC-HDP sont définis par:

Niveau 3 : Comportement	Niveau 2 : Activité	Niveau 1 : Atome d'activité
$\beta^{(3)} \sim GEM(\gamma^{(3)})$ $\pi^{(3)} \sim DP(\alpha^{(3)}, \beta^{(3)})$ $z_t^{(3)} \sim \pi^{(3)}$	$\beta^{(2)} \sim GEM(\gamma^{(2)})$ $\pi_l^{(2)} \sim DP(\alpha^{(2)}, \beta^{(2)})$ $z_{tj}^{(2)} \sim \pi_{l=z_t^{(3)}}^{(2)}$	$\beta^{(1)} \sim GEM(\gamma^{(1)})$ $\pi_\ell^{(1)} \sim DP(\alpha^{(1)}, \beta^{(1)})$ $z_{tji}^{(1)} \sim \pi_{\ell=z_{tj}^{(2)}}^{(1)}$ Trajectoires (signatures) $\phi_k \sim H(\lambda)$ $x_{tji} \sim F(\phi_{k=z_{tji}^{(1)}})$

où le vecteur α de paramètres d'intensités $(\alpha^{(1)}, \alpha^{(2)}, \alpha^{(3)})$ régule le nombre de pics. Afin de clarifier le lien entre le modèle et le jeu de données, la figure 10 montre comment le jeu de données hiérarchiquement structuré est lié aux différentes couches du modèle. Les notations B_n , a_n et aa_n correspondent aux respectivement aux clusters de *comportements*, *activités* et d'*atomes d'activités*.

3.3. Critère d'arrêt

Le MLC-HDP explore de manière itérative le domaine des solutions avec une part d'aléatoire (approche de type Monte-Carlo, cf. construction "stick breaking", et les différentes phases d'échantillonnage). Ainsi, la méthode converge généralement vers

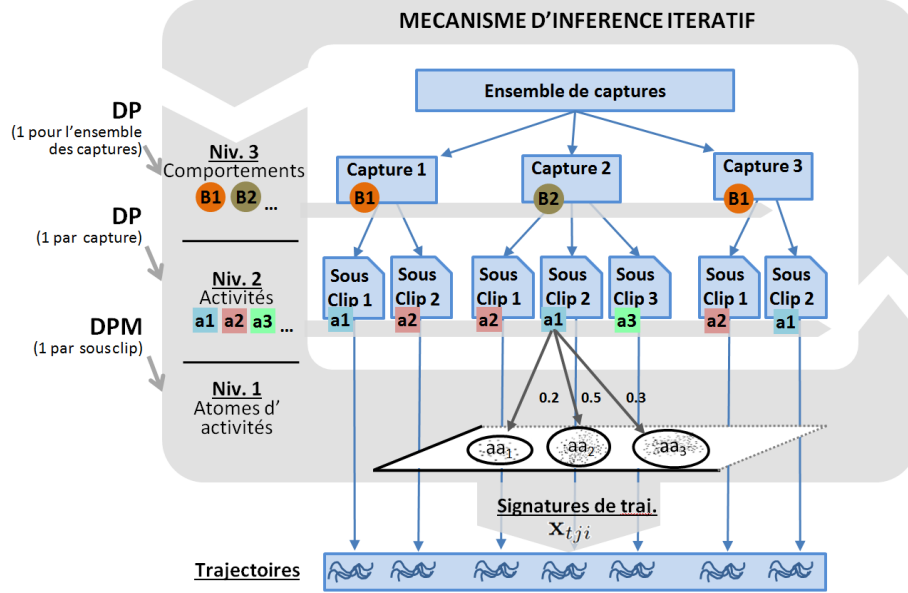


Figure 10. Illustration du principe d'inférence des paramètres du modèle à partir des données. Les boîtes bleues indiquent la structure du jeu de données. Le cycle grisé représente le principe de modélisation des données selon plusieurs niveaux. Les paramètres du modèle MLC-HDP (β, π, z) s'adaptent au fur et à mesure des itérations pour expliquer au mieux les données.

une solution optimum locale. L'idée est alors de laisser évoluer le MLC-HDP assez longtemps et de retenir la solution (modèle de mélange de Gaussiennes) qui maximise la vraisemblance par rapport aux données. Un indicateur classique est la log vraisemblance négative que l'on cherche à minimiser :

$$\log p(X|\pi, \mu, \Sigma) = - \sum_{n=1}^N \log \left\{ \sum_{k=1}^K \pi_{n,k} N(x_n | \mu_k, \Sigma_k) \right\}.$$

avec $\pi_{n,k}$ la probabilité que la donnée x_n appartienne au cluster k (i.e. poids issu de l'équation de niveau 1 du MLC-HDP), et N la fonction de vraisemblance d'une loi normale multi-variée. Avec μ_k et Σ_k les paramètres de la Gaussienne modélisant le cluster k . La figure 11 donne un aperçu de l'évolution de la log vraisemblance négative au fil des itérations du MLC-HDP. Nous avons constaté que les découvertes de nouveaux clusters pertinents se traduisent par des chutes soudaines de la log vraisemblance négative. Il serait également possible de définir un critère d'arrêt à partir d'un seuil de stabilité de la vraisemblance sur les n dernières itérations. Nous n'avons pas encore implémenté dans le cadre de nos travaux cette seconde idée de critère d'arrêt.

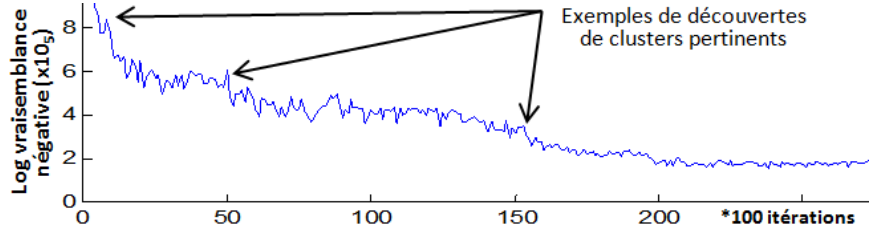


Figure 11. Progression de la log vraisemblance négative au cours des itérations. Les chutes notables correspondent à la découverte de clusters pertinents.

3.4. Signatures de trajectoires

Cette section propose une signature permettant de représenter de manière discriminante des trajectoires définies en 3D. Le but de nos travaux n'étant pas la recherche de signatures optimales pour la représentation de trajectoires, mais bien de trouver une signature fonctionnelle pour la mise en œuvre du MLC-HDP. Le lecteur souhaitant approfondir le domaine peut consulter les travaux de Yang *et al.* (2011). Par exemple, Beyan, Fisher (2013) représentent des trajectoires de poissons individuellement par 179 caractéristiques (originellement 776 caractéristiques extraites, réduites par ACP (Analyse en Composantes Principales)). La signature que nous proposons par la suite s'inspire en partie de ces travaux, notamment en considérant un mélange de deux types de caractéristiques dites "invariantes" et "globales".

En pratique, les trajectoires d'êtres vivants sont plutôt erratiques et nécessitent ainsi pour être qualifiées, un nombre important de caractéristiques. Contrairement à l'article mentionné précédemment, nous nous contentons d'un nombre plus modeste de caractéristiques, car la précision des clusters de trajectoires n'est pas primordiale dans notre approche. Rappelons que notre objectif est en premier lieu la classification de *captures* et *sousclips* et non la détection de trajectoires anormales. En fait, il est difficilement envisageable d'étudier les trajectoires de façon individuelle car il est compliqué de caractériser ce qu'est une trajectoire normale. Nous nous intéressons donc plutôt à des comportements potentiellement anormaux de cohortes. La détection de trajectoires anormales d'un point de vue individuel nécessiterait de repenser à une signature éventuellement plus riche.

Soit pour chaque trajectoire, un vecteur constitué des 7 caractéristiques invariantes ainsi que de 2 caractéristiques globales précisées ci-dessous. En aval, à partir de l'ensemble des signatures extraites, on normalise (i.e. recentrage et division par l'écart-type) les vecteurs sur chacune de leurs dimensions.

Caractéristiques invariantes : la Fonction de Distance Centrée (CDF) mentionnée par Bashir *et al.* (2006) consiste à calculer la distance relative entre chaque point de la trajectoire et son centroïde, défini par $d_c(a) = \sqrt{(X_a - \bar{X})^2 + (Y_a - \bar{Y})^2 + (Z_a - \bar{Z})^2}$ avec a l'index du point de la trajectoire (cf. Figure 12). Les 7 caractéristiques suivantes sont extraites à partir du vecteur d_c : la moyenne, l'écart-type, le

minimum, le maximum, l'asymétrie, le coefficient d'aplatissement et le nombre de croisements à la moyenne (en analogie avec le concept de "zero-crossing").

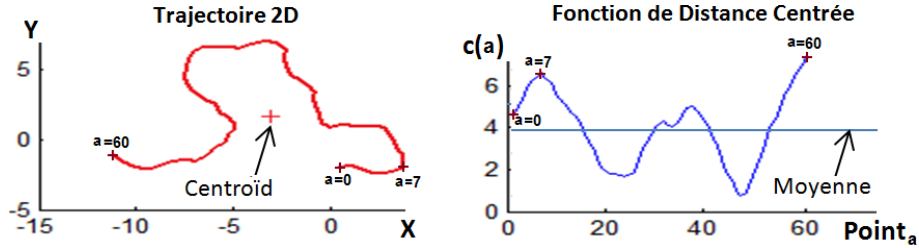


Figure 12. Exemple de calcul de Fonction de Distance Centrée (CDF) à partir d'une trajectoire 2D.

Caractéristiques globales : les angles moyens de la trace par rapport à l'horizontale et la verticale, ainsi que la distance moyenne à la ruche.

En conclusion, chaque trajectoire (ou observation) x_{tji} est exprimée par sa signature dans $\mathbb{R}^9$ avec t l'indice de la *capture* dans le corpus, j l'indice du *sousclip* dans la *capture* et i l'indice de la trajectoire dans le *sousclip*. Notons que l'on ne s'occupe pas de l'ordre temporel des trajectoires au sein de chaque *sousclip*.

Dans la partie expérimentation qui va suivre, nous utilisons une implémentation du MLC-HDP en Matlab (ainsi que certaines parties critiques en langage C). Nous nous sommes appuyé sur l'exemple mis à disposition par Dr. Wulsin, qui s'est lui-même appuyé sur le code du HDP fourni par Yeh Whye Teh.

4. Validation de l'approche par Simulation Multi-Agents (SMA)

4.1. Pertinence du couple SMA / MLC-HDP

L'avantage de la validation par un SMA est double. Elle permet principalement de conserver notre quête de minimisation de l'*a priori* aussi bien dans la simulation des données que dans la modélisation de celles-ci.

Le premier avantage : les SMAs permettent de simuler des *comportements* variés à partir de modèles simples comportant peu de paramètres. On peut ainsi générer des vérités terrain sans avoir recours à un processus d'annotation laborieux. Nous le verrons par la suite, les *comportements* (concepts de haut niveau) peuvent par exemple être observés au travers de variables d'environnement (sachant que les agents agissent sur leur environnement). Quant aux trajectoires (concepts de bas niveau), elles sont directement engendrées par le déplacement des agents mobiles.

Le second avantage : les trajectoires simulées ne sont pas directement contraintes par un modèle défini de dynamique. En effet, il ne serait pas judicieux de chercher à modéliser des données dont on connaît et maîtrise en détails les paramètres permettant de les simuler. Contrairement aux scènes encombrées fréquemment rencontrées

en vidéosurveillance (e.g. trafic, piétons), l'évolution d'insectes ou d'animaux en général donne lieu à des scènes bien moins structurées. Cela s'explique par le peu de contraintes engendrées par un environnement ouvert (e.g. scène non définie par des murs, couloirs, routes). Il arrive souvent même que les cibles vivantes ne suivent aucune tendance perceptible de mouvement. Ce phénomène est aisément simulable à l'aide d'un SMA.

4.2. Génération du jeu de données structuré de trajectoires et de la vérité terrain associée

Lors d'une simulation SMA, chaque agent produit un nombre variables de trajectoires. On considère comme "trajectoire" la séquence de points issue du déplacement d'un agent entre un point d'intérêt A (e.g. nids, ruche) et un point d'intérêt B (e.g. source de nourriture, limite de la zone d'observation). Ces trajectoires sont collectées et rassemblées dans un jeu de données. Celui-ci est organisé selon la structure hiérarchique en "*captures / sousclips*" décrite dans la section 3.1. Simultanément, on peut observer certaines variables environnementales, qui serviront de vérité terrain quant à l'évolution du comportement de la colonie dans le temps. La figure 13 illustre comment les trajectoires sont collectées et comment des variables d'environnement peuvent témoigner du comportement de la colonie.

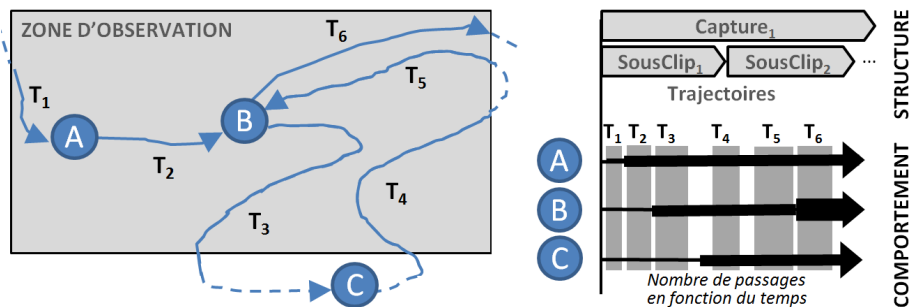


Figure 13. À gauche : schéma des trajectoires engendrées par un agent à l'issue d'une simulation SMA. La collecte des données est restreinte par une zone d'observation. À droite : structure des données de trajectoires et observation du comportement au travers de variables environnementales (e.g. nombre d'arrivées aux points de référence en fonction du temps).

Niv. 3 - Captures & comportements : une *capture* correspond à une simulation complète. Celle-ci est lancée sur un jeu de paramètres et engendre un *comportement* particulier. Prenons l'exemple d'une simulation SMA dont la durée est arbitrairement fixée à 3000 itérations, soit le temps potentiellement nécessaire pour que la colonie d'agents enchaîne un certain nombre d'*activités* différentes. Le nombre d'itérations (i.e. pas de temps à simuler) varie en fonction du modèle SMA et des paramètres de simulation. Par exemple, on peut décider d'arrêter la simulation dès lors que le

SMA arrive à un certain état d'équilibre (e.g. ressources de l'environnement totalement consommées).

Niv. 2 - Sousclips & activités : les *captures* sont arbitrairement séparées en 32 *sousclips*. Ce qui donne des *sousclips* de $3000/32 = 92$ itérations, d'une durée suffisante pour englober un nombre significatif de trajectoires, mais pas trop pour éviter d'avoir plusieurs *activités* différentes par *sousclip*.

Niv. 1 - Signatures de trajectoires : à partir des trajectoires individuelles sont calculées des signatures (cf. Section 3.4) qui serviront de données en entrée du MLC-HDP.

Dans notre cas, un modèle SMA paramétré par un jeu de paramètres fera émerger à l'issue de différentes instances de simulation des comportements proches, voir similaires. De légères variations sont tout de même observées quant aux instants et aux vitesses de convergence des différentes activités (e.g. récolte d'une source, recherche de nourriture).

4.3. Étude à partir d'un modèle de colonie des abeilles

Chiron *et al.* (2014) ont proposé une première étude s'appuyant sur une implémentation en 2D d'un premier modèle de simulation de colonie d'abeilles à la recherche de nourriture (cf. implémentation proposée dans les "matériaux supplémentaires" de l'article cité). Ce modèle est inspiré des règles de communication entre abeilles exploitées par Figueroa *et al.* (2009). Dans la nature, par exemple, l'échange d'information se fait grâce à une danse effectuée au sein de la ruche. Notre modèle de colonie d'abeilles fait émerger des comportements grâce à ce partage d'information entre individus.

Dans ce présent article, nous allons plus loin avec une implémentation d'un modèle similaire mais en 3D cette fois. Quelques améliorations ont été apportées dans le but de rendre la simulation plus réaliste. Notamment, les agents (abeilles) génèrent des trajectoires plus bruitées (e.g. incertitude de la direction prise par les abeilles en fonction de la distance à leur destination) et générant des comportements plus complexes (e.g. retour indirecte à la ruche au bout d'un laps de temps donnée). L'implémentation et les simulations sont effectuées sous Netlogo.

4.3.1. Détails sur le modèle multi-agents d'abeilles

Voici en quelques lignes une présentation du fonctionnement de notre modèle de colonie d'abeilles. Chaque abeille de la colonie est représentée par un agent. Une abeille peut prendre tour à tour les 3 statuts suivants (cf. figure 14) :

- **Exploratrice** : explore le monde à la recherche d'une source de nourriture. Si l'exploration demeure infructueuse au delà d'un laps de temps donné, l'abeille est contrainte de revenir à la ruche (de manière indirecte). Lorsque qu'une source de nourriture est trouvée, l'abeille peut devenir "Exploitante";

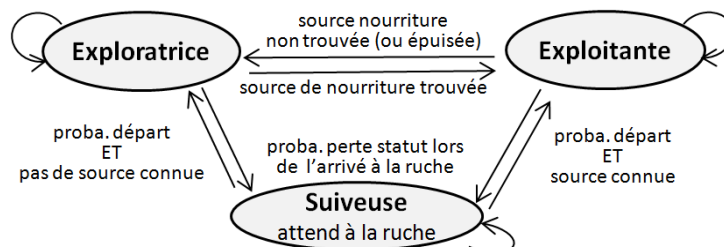


Figure 14. Automate modélisant les conditions de transition entre les 3 statuts que peuvent prendre les agents.

- **Exploitante** : exploite une source de nourriture en faisant des allers-retours entre la source et la ruche. Lorsque la source est épuisée, l'abeille peut redevenir "Exploratrice" ou devenir "Suiveuse";
- **Suiveuse** : positionnée dans la ruche, elle accumule les informations sur les sources de nourriture partagées par les "Exploitantes" lors des multiples va-et-vient de celles-ci. Au bout d'une certaine période, la "Suiveuse" peut devenir "Exploitante" et exploiter à son tour la source de nourriture la plus partagée par ses congénères.

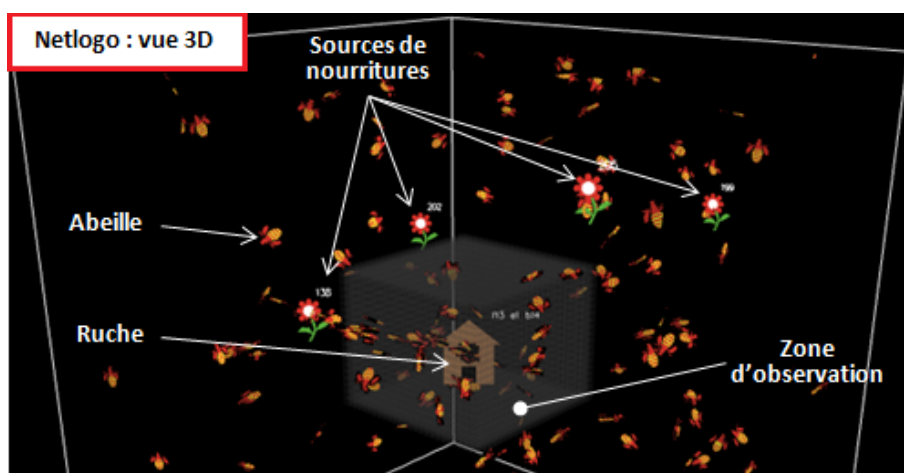


Figure 15. Simulation en cours (300ème itération) de notre modèle multi-agents de colonie d'abeilles. Le symbole "maison" au centre représente la ruche, les 4 fleurs représentent les 4 sources de nourriture, les éléments oranges/rouges symbolisent les abeilles et la zone grisée/transparente délimite la zone d'observation.

Nous proposons de nous focaliser sur l'émergence de comportements liés à des perturbations de la colonie (e.g. intoxication, prédation). Dans ce but, nous axons notre modèle autour de deux paramètres principaux que nous ferons varier et qui sous-tendent des changements comportementaux :

- TC : le taux de communication entre les abeilles au niveau de la ruche;

– TS : le taux de sensibilité (ou de détection) des abeilles vis-à-vis des sources de nourriture rencontrées.

D'autres paramètres de second plan sont fixés par défaut, par exemple : le nombre d'abeilles (ou population); la position de la ruche; le nombre, la position des sources de nourriture et la quantité associée; la probabilité de perdre son statut lors de l'arrivée à la ruche; la fréquence maximum des départs au niveau de la ruche. Dans notre modèle, plus la source de nourriture est éloignée de la ruche, plus l'abeille sera rapide et sa trajectoire bruitée. Ceci modélise notamment l'incertitude de la direction prise au départ de la ruche et permet, par la même occasion, de générer des trajectoires aux caractéristiques intrinsèques variées. La figure 15 illustre ce modèle en action. La zone d'observation est restreinte à un périmètre limité autour de la ruche. Nous le verrons plus tard dans la section 5, ceci est en adéquation avec les conditions réelles, pour lesquelles, de fait, les trajectoires sont observées uniquement lorsqu'elles passent dans le champ de la caméra, elle même placée à proximité de la ruche.

4.3.2. Résultats de clustering MLC-HDP - Niveau 2

Nous nous intéressons en premier lieu aux résultats de la modélisation MLC-HDP sur une seule *capture* limitée donc au niveau 2 "*activité*". La *capture* est obtenue à partir du modèle multi-agents d'abeilles défini précédemment. Dans l'exemple qui suit, les paramètres utilisés pour la simulation SMA sont : population=200, TS=80, TC=80, quatre sources de nourriture disposées à une distance croissante de la ruche. La *capture* dure 2365 itérations (ou unités de temps), est segmentée en 30 *sousclips*, et comporte en tout 4058 trajectoires. La figure 16 montre les données de simulation de la capture (i.e. graphique d'évolution des réserves de nourriture), et les résultats de modélisation MLC-HDP au niveau 2. L'interprétation visuelle des étapes de la chronologie démontre la pertinence des résultats de clustering. En effet, chaque *sousclip* a été associé à une des 11 *activités* distinctes recensées par le MLC-HDP. Ces 11 *activités* s'appuient sur 16 *atomes d'activité* auxquels sont associés différents poids. Comme le montre la figure 16, les transitions entre les *activités* sont en phase avec des événements notables s'étant produits lors de la simulation. Par exemple, sur la figure, la transition entre les phases "Récolte rapide de S₄" et "Libre" est nette, on distingue bien en effet une rupture des étiquettes (jaunes et marron) associées aux *sousclips* concernés. Il est important de remarquer que la granularité du jeu de données (e.g. longueur des *sousclips*) peut influencer la qualité des résultats. De fait, des événements notables peuvent se mélanger au sein d'un même *sousclip* et donner naissance à des *activités* de nature différente.

La figure 17 illustre les résultats de clustering des signatures de trajectoires en "*atomes d'activité*" (niv. 1). Sur cette exemple, on remarque la capacité du MLC-HDP (grâce aux DP et DPMs) à expliquer les données par un mélange de lois normales multivariées (représentées sur la figure par des Gaussiennes tridimensionnelles). Aux quatre Gaussiennes les plus importantes sont associées des trajectoires dites de "bruit" (i.e. ne correspondant à aucune activité clairement distinguable), ce qui correspond probablement à des trajectoires d'abeilles en mode "exploration". En revanche les

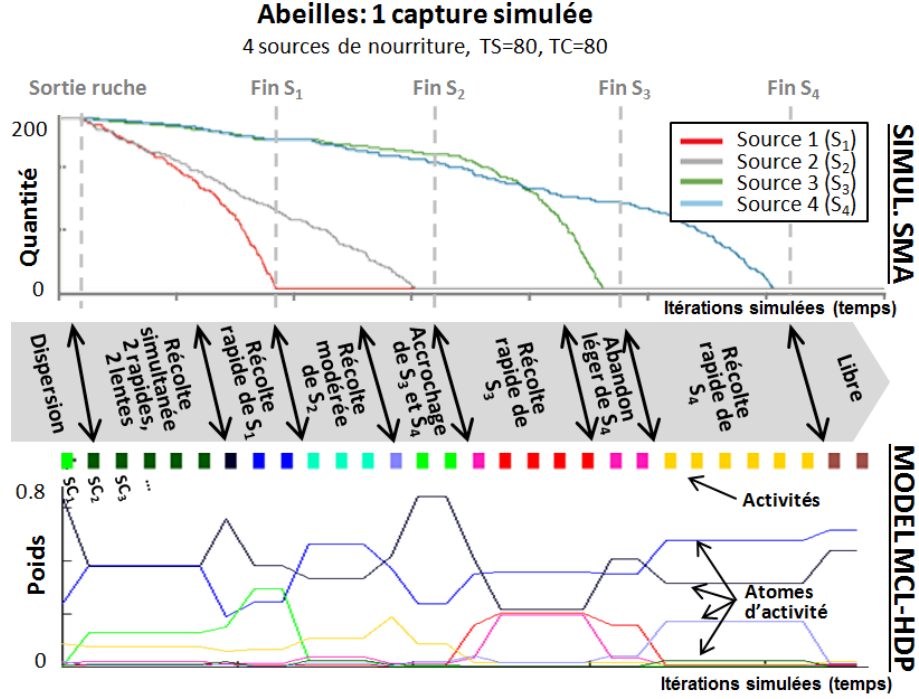


Figure 16. En haut, graphiques d'évolution des sources de nourriture. En bas, résultats du clustering MLC-HDP (niv. 2). Au centre, 9 interprétations à la série d'activités obtenues par le MLC-HDP. Le code couleur des "atomes d'activité" est commun à la figure 17, où ils sont représentés par des clusters sous forme d'ellipsoïdes.

supports des autres Gaussiennes correspondent aux zones où la concentration en signatures de trajectoires est forte, et donc l'activité associée est moins chaotique. On note d'ailleurs que les paramètres des Gaussiennes (μ et σ) couvrent un domaine large.

4.3.3. Résultats du clustering MLC-HDP - Niveau 3

Nous testons dans cette section la capacité du MLC-HDP à regrouper un grand nombre de *captures* en fonction de leur *comportement* (niv. 3). Pour cela, nous avons simulé 80 *captures* à partir de 4 configurations de paramètres différents, soit 20 *captures* par configuration. Voici les 4 jeux de paramètres que nous avons fait varier dans notre modèle :

- **Configuration A (TC=95, TS=35)** : récolte lente car peu sensible aux sources de nourriture mais convergence (i.e. effort de la colonie focalisée sur une ou plusieurs tâches utiles) facilitée par un taux de communication élevé.
- **Configuration B (TC=1, TS=35)** : récolte lente car peu sensible aux sources de nourriture et aucune convergence vers l'une ou l'autre des sources.

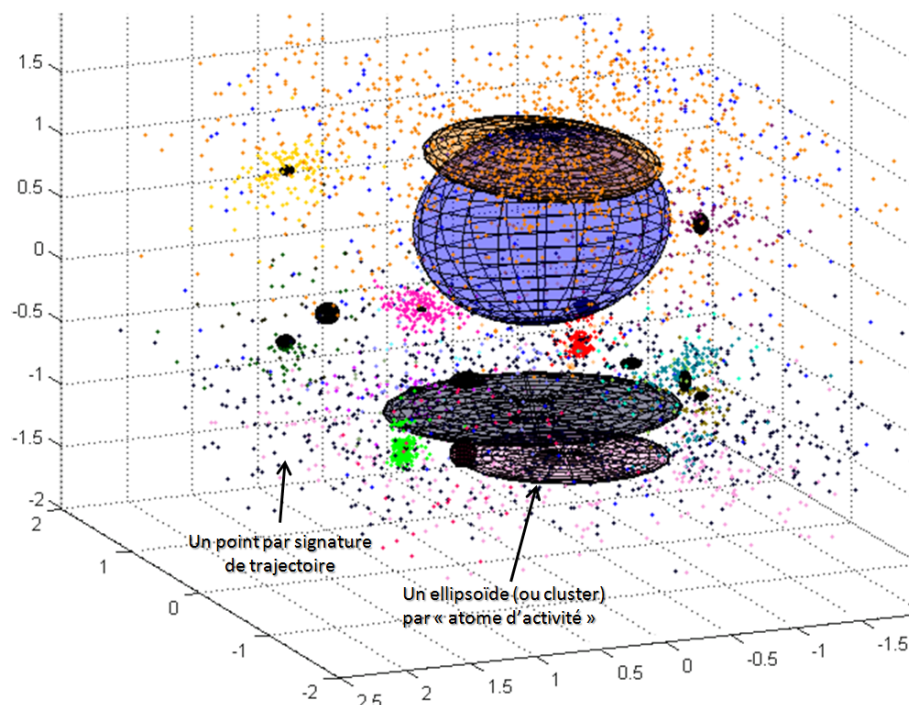


Figure 17. Résultat de clustering (niv. 1) sur l'ensemble des signatures de trajectoires contenues dans une capture. Les 16 ellipsoïdes représentent les clusters, et la couleur indique le cluster d'appartenance des points. Une ACP a été appliquée sur les données de manière à réduire le nombre de dimensions à trois afin de permettre une visualisation pertinente.

- **Configuration C (TC=1, TS=95)** : récolte rapide car très sensible aux sources de nourriture, mais aucune convergence vers l'une ou l'autre des sources.
- **Configuration D (TC=95, TS=95)** : récolte rapide car très sensible aux sources de nourriture et convergence facilitée par un taux de communication élevé.

Les paramètres communs aux 4 configurations sont une population de 200 individus et 2 sources de nourriture.

La figure 18 illustre à la fois le jeu de données (vérités terrains associées aux 80 captures) et les résultats de clustering du MLC-HDP sur les *comportements* (niv. 3). Cette figure reprend la notion de courbe d'évolution des réserves de nourriture (c.f. figure 16), de manière simplifiée et sur un plus grand nombre d'exemples. La partie inférieure de la figure 18 montre une synthèse des résultats de modélisation comportementale par MLC-HDP basée les données de trajectoires. Notre hypothèse est que les données de trajectoires reflètent d'une certaine manière le même comportement global que nous pouvons observer au travers de l'évolution des sources de nourriture.

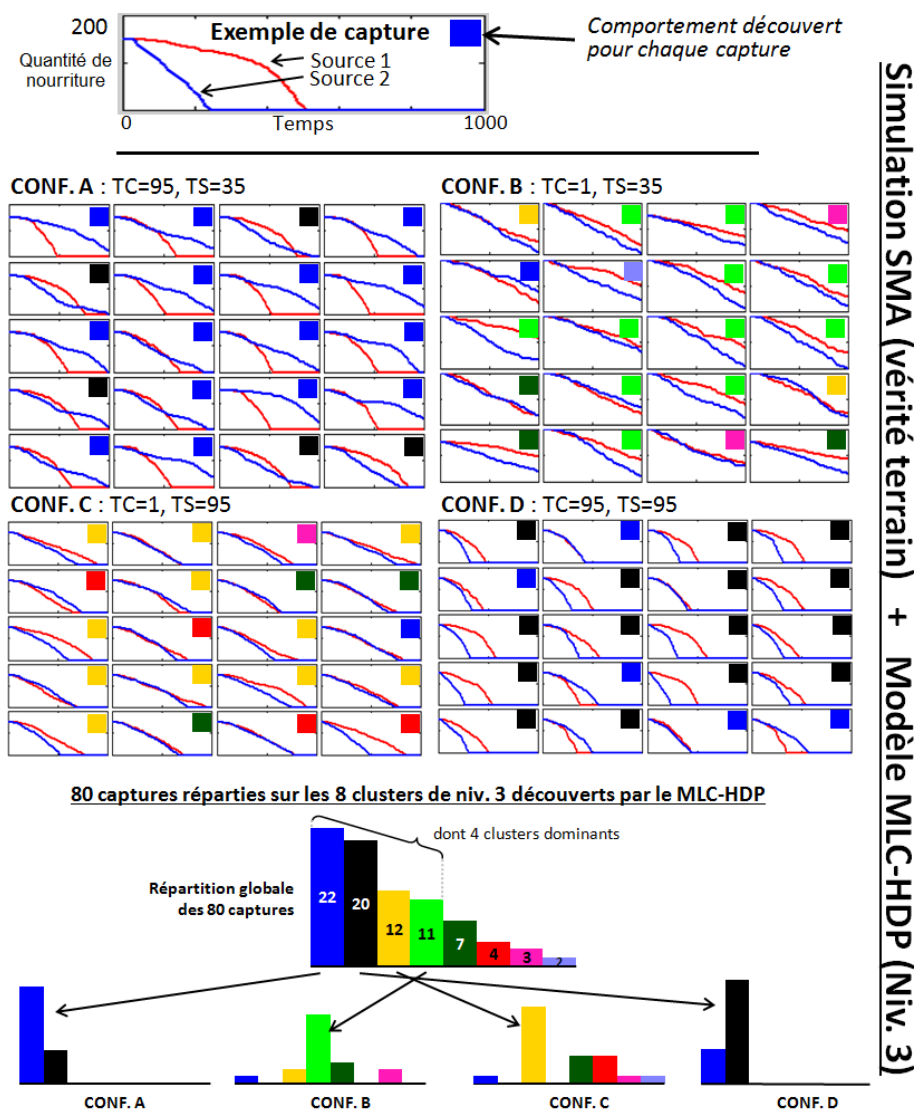


Figure 18. Données de simulation multi-agents "Abeilles" et résultats du clustering MLC-HDP (niv. 3). La partie supérieure de la figure illustre comment le graphique d'évolution des réserves de nourriture qualifie chaque capture. En dessous, est présentée la vérité terrain comportementale (jeux de paramètres & graphiques d'évolution des réserves de nourriture) pour les 80 captures générées à partir de notre modèle multi-agents de colonie d'abeilles. Puis, la partie inférieure présente les résultats de la modélisation MLC-HDP au niveau 3, comportement, à partir des 224 309 trajectoires structurées en captures.

En termes de résultat, le MLC-HDP a découvert 8 *comportements* différents (clusters de niv. 3), 16 *activités* (clusters de niv. 2) et 6 *atomes d'activité* (clusters de trajectoires, niv. 1). Parmi les 8 *comportements*, on perçoit clairement 4 *comportements* majeurs qui correspondent aux 4 jeux de paramètres utilisés pour la simulation des *captures*. Les 4 autres *comportements* représentés de façon minoritaire permettent probablement de modéliser les comportements "déviant" issus de la grande variabilité du jeu de données. En effet, notre modèle de colonie d'abeilles génère un certain nombre de comportements qui ne convergent pas comme la majorité des simulations lancées sur un même jeu de paramètres. La vérité terrain associée au jeu de données de trajectoires n'est donc pas parfaite, mais cela fait partie des difficultés que nous souhaitons relever avec notre approche.

Nous utilisons dans cette expérience un jeu de données constitué de 224 309 signatures de trajectoires. Celui-ci est structuré en *sousclips* et *captures* auxquels sont associés des Processus de Dirichlet. Ce qui revient à échantillonner 2480 DP par itération. Nous avons lancé le MLC-HDP sur 810 itérations, ce qui correspond à environ 14h de calcul sur une machine équipée d'un Intel Core i3 à 2.5 Ghz.

5. Expérimentation sur trajectoires réelles d'abeilles

Nous testons notre approche de modélisation multi-échelles sur des données réelles. L'expérience présentée dans cette section s'inscrit dans le cadre d'études menées en collaboration avec l'INRA Magneraud et le Muséum National d'Histoire Naturelle (MNHN). Celle-ci vise à identifier des comportements types au sein d'une colonie d'abeilles en s'appuyant sur les trajectoires de vol des individus devant la ruche.

5.1. Collecte de trajectoires

La figure 19 illustre le dispositif d'acquisition utilisé pour appréhender l'environnement en 3D et collecter les trajectoires des abeilles en vol (cf. chaîne complète détaillée par Chiron *et al.* (2013)). Le trépied servant à tenir la caméra est fixé sur un toit de ruche mobile. Cela permet de conserver la même configuration (inclinaison, hauteur de la caméra) lors des déplacements. Sur des ruches standards, les séquences vidéo sont ainsi collectées dans un même référentiel.

En résumé :

- **Dispositif d'acquisition** : en réponse aux contraintes fortes liées aux conditions d'acquisition en milieu naturel, nous utilisons une caméra stéréovision haute fréquence (TYZX G3 Evo, 752×480 px, @50 fps) pour appréhender en 3D la scène encombrée. La caméra, fixée sur le toit de la ruche, observe les abeilles en vol devant l'entrée dans volume inscrit dans une pyramide de $(50 \times 40) \times 50$ cm.
- **Détection des cibles** : les abeilles en vol sont détectées et localisées en 3D grâce à une méthode nommée "Segmentation HIDS", mettant en œuvre une fusion optimale

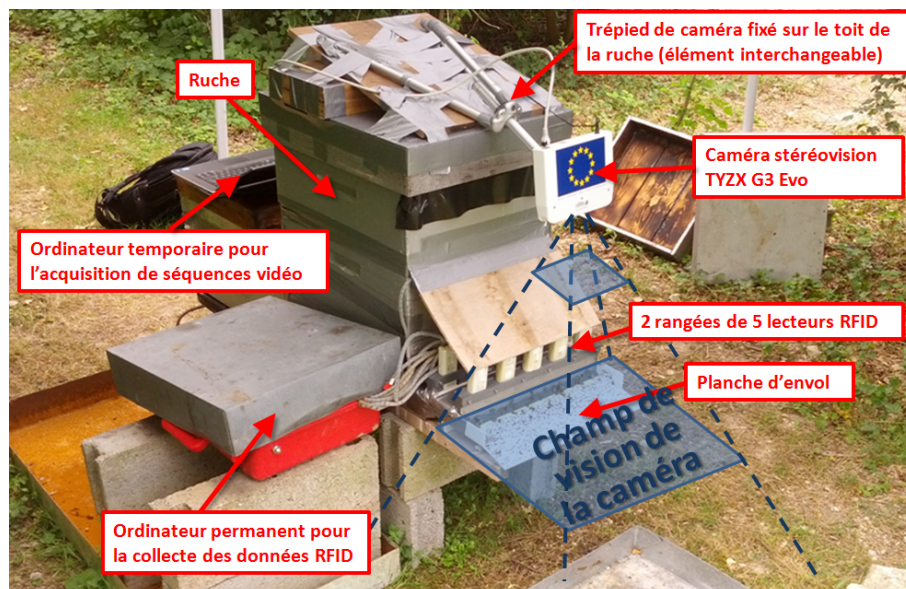


Figure 19. Dispositif d'acquisition stéréovision à l'entrée de la ruche.

des informations d'intensité, de profondeur et de mouvement. On peut observer des séquences contenant jusqu'à 25 abeilles par image.

– **Suivi multi-cibles** : les trajectoires sont calculées à l'aide d'un couple "filtre de Kalman / Global Nearest Neighbor" permettant respectivement d'estimer et de prédire la position des cibles détectées suivant un modèle de dynamique et d'apparier selon un minimum local les pistes suivies et les observations.

Au total, avec l'aide des biologistes, nous avons collecté sur le terrain 74 captures RGB-D d'une durée de 15 minutes environ, soit au total plus de 20 heures de vidéos d'abeilles en vol réparties sur le mois de juillet 2013. La période estivale coïncide avec une forte activité des colonies, notamment due aux opérations de butinage. À partir de cet ensemble de vidéos, environ 30 millions d'abeilles ont été détectées et localisées en 3D, puis plus de 800 000 trajectoires d'abeilles ont été assemblées. Une étape de filtrage (i.e. traces trop courtes ou physiquement incohérentes) réduit le corpus à environ 300 000 trajectoires. Ces trajectoires ont été regroupées suivant la structure *capture/sousclip* décrite précédemment.

5.2. Modélisation comportementale multi-échelle

La figure 21 montre les résultats de la modélisation MLC-HDP au niveau 3 *comportements* sur les 74 *captures*. En guise d'exemple pour la lecture, une *capture* est détaillée dans la partie supérieure de la figure. Ici, ce ne sont pas les détails au ni-

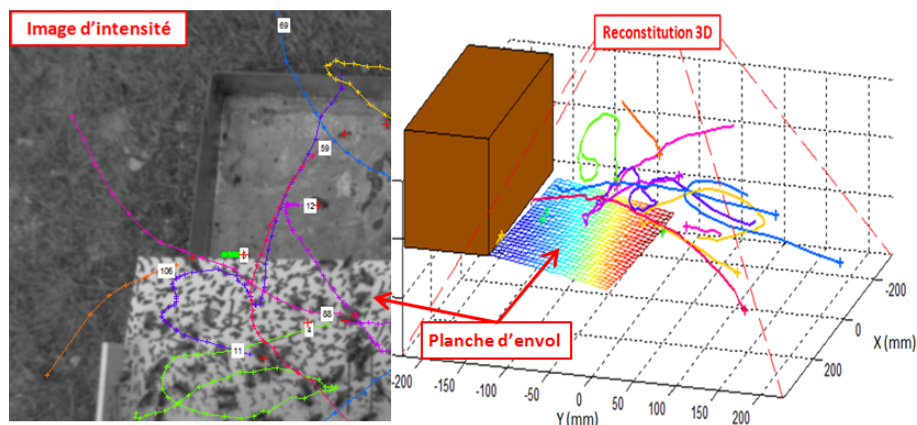


Figure 20. À gauche : image de l'entrée de la ruche capturée par la caméra placée sur le toit. À droite : reconstitution 3D de la scène (ruche, planche d'envol et trajectoires des abeilles).

veau des *sousclips* qui sont intéressants, mais bien le *comportement* associé à chaque *capture*.

Le premier constat est que l'on observe une certaine stabilité comportementale au cours du temps. Notamment, on retrouve peu de *comportements* isolés. Parfois même, les *comportements* de débuts d'après-midi sont similaires d'une journée sur l'autre (e.g. du 02/07 au 12/07).

Le deuxième constat est que notre modélisation par MLC-HDP semble robuste aux variations du nombre de trajectoires par *sousclip* (cf. histogrammes gris). En effet, on observe par exemple un comportement constant sur la journée du 25/06 (sauf à 16h et 17h), malgré une variation importante du nombre de trajectoires collectées. Ce nombre est corrélé au nombre d'abeilles en vol devant la ruche.

Il est difficile de qualifier ces résultats d'un point de vue biologique. En effet, le peu de données haut niveau (c'est à dire autre que les trajectoires) dont nous disposons (e.g. historique de la météo, histoire de vie des abeilles individualisée), de part leur faible robustesse, ne nous ont pas permis à l'heure actuelle de fournir une explication biologique fiable ou d'observer une corrélation avec les résultats trouvés par notre modélisation MLC-HDP. Nous travaillons actuellement avec les biologistes de l'INRA pour améliorer les protocoles expérimentaux intégrant des informations de plus haut niveau.

A titre indicatif, ce sont 1076 DPs (74 *captures* composées en moyenne de 13 *sousclips*) qui sont échantillonnés à chaque itération du MLC-HDP. Après avoir laissé itérer l'algorithme quelques heures, il s'est avéré que la meilleure solution (minimisant la log vraisemblance négative) a été atteinte à la 46^{ème} itération.

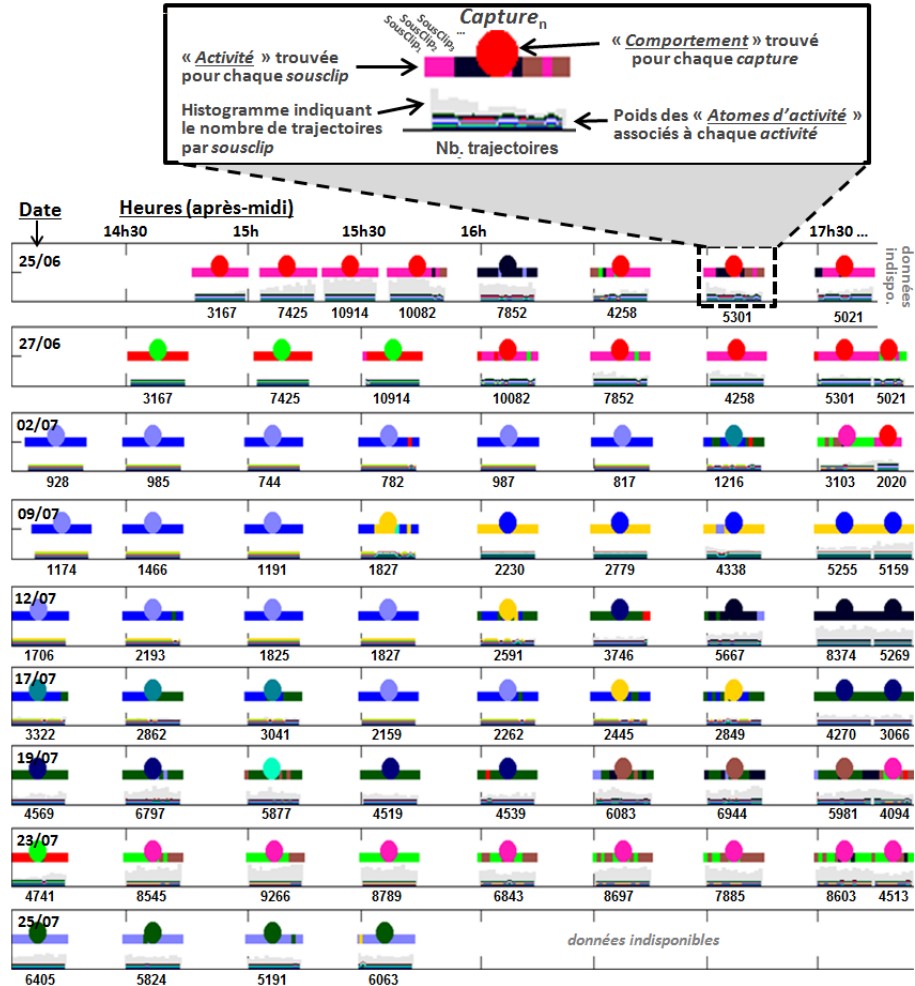


Figure 21. Résultat de la modélisation MLC-HDP (niv. 3 - comportements) sur l'ensemble des 74 captures acquises durant l'été 2013 à l'INRA Magneraud.

6. Conclusion

6.1. Approche méthodologique

Dans ce chapitre, nous avons proposé une approche pour la découverte et la classification de comportements à partir d'un ensemble structuré de trajectoires selon différentes échelles temporelles. L'intérêt principal de l'approche est qu'elle ne nécessite pas d'injection de connaissances métier. Et finalement c'est la structure de données elle-même qui fournit l'*a priori* nécessaire pour l'élaboration du modèle.

Ensuite, dans ce contexte de minimisation de l'*a priori*, nous avons proposé une validation à l'aide de modèles multi-agents. Les expériences menées à partir de données simulées ont confirmé l'intérêt d'une étude comportementale multi-niveaux sur des données structurées hiérarchiquement.

Finalement, les contributions apportées dans cet article sont les suivantes :

- l'utilisation d'un HDP multi-niveaux pour la découverte de comportements émergents. À notre connaissance, les approches de type HDP, en relation avec la modélisation comportementale, utilisent au maximum 2 niveaux;
- l'utilisation de caractéristiques extraites de trajectoires individuelles éparses, plutôt qu'une approche fondée sur le flot optique telle que traditionnellement adoptée pour les scènes encombrées;
- la validation originale par simulation système multi-agents, fournissant indirectement la vérité terrain sémantique associée au jeu de données de trajectoires.

En termes de perspective méthodologique, il serait intéressant de prendre en compte l'évolution des trajectoires dans le temps (ou plutôt leurs caractéristiques) en étudiant les probabilités de transition aux niveaux des *activités* et des *comportements*. Dans ce contexte, les méthodes de type HDP-HMM (pour HDP Hidden Markov Models) sont une piste à explorer.

6.2. Application sur des données réelles

Nous avons confronté notre méthodologie à des données de trajectoires réelles dans le cadre d'une application de surveillance apicole. Ce travail correspond à une étude de faisabilité pour les projets interdisciplinaires bioinformatiques EPERAS et RISQAPI (en collaboration avec l'INRA et le MNHN). Ces projets ont pour objectif d'étudier les changements comportementaux des abeilles face à l'évolution des conditions de leur environnement (e.g. stress dû à la prédation par le frelon asiatique, intoxication par des produits phytosanitaires assimilés à des doses sublétales). Les résultats de modélisation obtenus semblent cohérents, de part leur stabilité relative, et démontrent ainsi à nouveau la pertinence de l'approche proposée dans cet article.

Concernant les perspectives applicatives, il serait intéressant d'aller plus loin en construisant, dans la mesure du possible, une vérité terrain réelle à l'aide des informations fournies par le dispositif d'identification RFID placé à l'entrée de la ruche (cf. Figure 19). Ce dernier permettrait d'associer aux trajectoires des informations sur le degré d'intoxication des abeilles et leur histoire de vie. Ces travaux, nécessitant un effort important d'ingénierie, sont en cours de développement au L3i.

Remerciements

Nous tenons à remercier Dr. Drausin Wulsin pour la mise à disposition d'un exemple d'implémentation de l'approche MLC-HDP (cf. Wulsin *et al.*, 2012) et pour nous avoir fait bénéficier de son expertise vis-à-vis de notre application. Merci aux biologistes de

l'INRA et du MNHN pour nous avoir accompagné lors des différentes campagnes de captures vidéo dans leurs ruchers expérimentaux. Ce travail a été financé par le FEDER (Fond pour le Développement Européen pour la Région, contrat: 35053) et la région Poitou-Charentes.

Annexe

Loi multinomiale / Loi de Dirichlet

La loi binomiale ne prend en compte que deux issues possibles, ce qui est souvent illustré par l'exemple de la pièce de monnaie tombant sur "pile" ou "face". La loi multinomiale est sa généralisation permettant la prise en compte plusieurs résultats. Elle peut être illustrée par n jets d'un dé à K faces, où chaque face i a une probabilité p_i de succès, et où la somme des probabilités p_i est égale à 1. La densité de la loi multinomiale est notée :

$$f(x|n, p) = \frac{n!}{x_1! \dots x_K!} p_1^{x_1} \dots p_K^{x_K}.$$

où $x = (x_1, \dots, x_K)$ est le nombre de succès de chaque face pour n tirages suivant les probabilités $p = (p_1, \dots, p_K)$ relatives à chaque face du dé. En retournant le problème, on cherche à estimer la densité de probabilité des p_i à partir des n observations. On utilise alors la loi de Dirichlet (généralisation de la loi Beta) qui est la conjuguée de la loi multinomiale, où $\alpha = (\alpha_1, \dots, \alpha_K)$ représente le nombre d'observations de chaque face. La densité de probabilité de la loi de Dirichlet est notée :

$$f(p_1, \dots, p_K | \alpha) = \frac{1}{Beta(\alpha)} \prod_{i=1}^K p_i^{\alpha_i - 1}.$$

avec $p_1, \dots, p_K$ les paramètres à estimer. On note la constante de normalisation $Beta$ exprimée à l'aide de la fonction Gamma.

$$Beta(\alpha) = \frac{\prod_{i=1}^K \Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^K \alpha_i)} \quad \text{et} \quad \Gamma(x) = \int_0^\infty e^{-t} t^{x-1} dt$$

Par la suite, on notera la loi de Dirichlet $Dir(\alpha_1, \dots, \alpha_K)$. La figure 22 illustre les fonctions de densité de la loi multinomiale et de sa conjuguée sur un exemple de dé à 3 faces. La loi multinomiale de paramètre $p = (0.5, 0.2, 0.3)$ pour $n = 10$ tirages, donne le maximum de densité de probabilité pour $x = (5, 2, 3)$ succès. De manière complémentaire, la loi de Dirichlet de paramètre $\alpha = (5, 2, 3) + 1$ donne le maximum de densité de probabilité pour $p = (0.5, 0.2, 0.3)$. L'idée est que, plus le nombre de tirages augmente, plus la densité vient se concentrer autour de ce maximum.

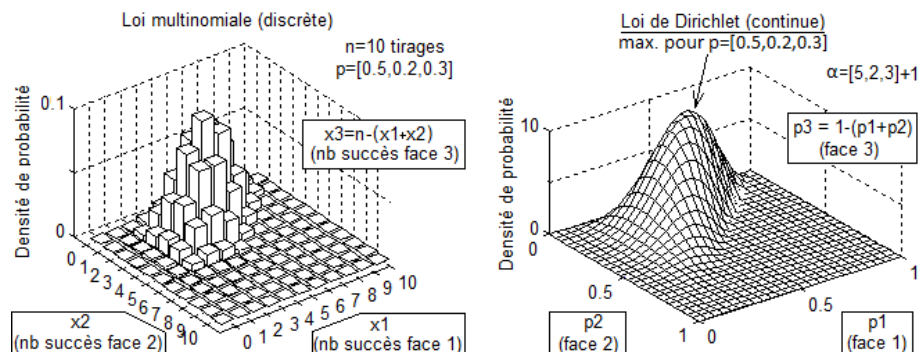


Figure 22. Exemple de fonctions de densité de probabilité : loi multinomiale et sa conjuguée, la loi de Dirichlet.

Bibliographie

- Bashir F., Khokhar A., Schonfeld D. (2006). View-invariant motion trajectory-based activity classification and recognition. *Multimedia Systems*, vol. 12, n° 1, p. 45-54.
- Beal M., Krishnamurthy P. (2006). Gene expression time course clustering with countably infinite hidden markov models. In *Proceedings of 22nd Conference on Uncertainty in Artificial Intelligence*.
- Beyan C., Fisher R. (2013). Detection of abnormal fish trajectories using a clustering based hierarchical classifier. In *Proceedings of the British Machine Vision Conference*. BMVA Press.
- Blackwell D., MacQueen J. B. (1973). Ferguson distributions via polya urn schemes. *The Annals of Statistics*, vol. 1, n° 2, p. 353-355.
- Chiron G., Gomez-Krämer P., Menard M. (2013). Detecting and tracking honeybees in 3d at the beehive entrance using stereo vision. *EURASIP Journal on Image and Video Processing*, vol. 2013, n° 1, p. 59.
- Chiron G., Gomez-Krämer P., Ménard M. (2014). Approche bayésienne non-paramétrique pour la découverte de comportements émergents à partir de trajectoires. In *Actes de la conférence RFIA 2014*. France.
- Fei-Fei L., Perona P. (2005). A bayesian hierarchical model for learning natural scene categories. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005*, vol. 2, p. 524-531 vol. 2.
- Feldman A., Balch T. (2004). Representing honey bee behavior for recognition using human trainable models. *Adaptive Behavior - Animals, Animats, Software Agents, Robots, Adaptive Systems*, vol. 12, n° 3-4, p. 241-250.
- Ferguson T. S. (1973). A bayesian analysis of some nonparametric problems. *The Annals of Statistics*, vol. 1, n° 2, p. 209-230.
- Figuerola J. R., Jacildo A. J., J.F R. (2009). Simulation of bee foraging behavior using biroi preference algorithm. In *Oral in the 15th BeeNet Conference and Techno-fora.*, p. 57-64.

- Fox E. B., Sudderth E. B., Willsky A. S. (2007). Hierarchical dirichlet processes for tracking maneuvering targets. In *Proceedings of the 10th International Conference on Information Fusion, 2007*, p. 1–8.
- Gorelick L., Blank M., Shechtman E., Irani M., Basri R. (2007). Actions as space-time shapes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, n° 12, p. 2247–2253.
- Griffin J. E., Steel M. F. J. (2006). Order-based dependent dirichlet processes. *Journal of the American Statistical Association*, vol. 101, n° 473, p. 179-194.
- Hongeng S., Nevatia R. (2001). Multi-agent event recognition. In *Proceedings of the 8th IEEE International Conference on Computer Vision, 2001*, vol. 2, p. 84-91 vol.2.
- Hospedales T., Gong S., Xiang T. (2009). A markov clustering topic model for mining behaviour in video. In *Proceedings of the 12th IEEE International Conference on Computer Vision, 2009*, p. 1165-1172.
- Intille S. S., Bobick A. F. (1999). A framework for recognizing multi-agent action from visual evidence. In *Proceedings of the 6th national conference on artificial intelligence and the 11th innovative applications of artificial intelligence conference innovative applications of artificial intelligence, 1999*, p. 518–525. Menlo Park, CA, USA, American Association for Artificial Intelligence.
- Khan Z., Balch T., Dellaert F. (2005). MCMC-based particle filtering for tracking a variable number of interacting targets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, n° 11, p. 1805-1819.
- Kuettel D., Breitenstein M., Van Gool L., Ferrari V. (2010). What’s going on? Discovering spatio-temporal dependencies in dynamic scenes. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2010*, p. 1951-1958.
- Morris B., Trivedi M. (2008). A survey of vision-based trajectory learning and analysis for surveillance. *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, n° 8, p. 1114-1127.
- Niebles J., Wang H., Fei-Fei L. (2008). Unsupervised learning of human action categories using spatial-temporal words. *International Journal of Computer Vision*, vol. 79, n° 3, p. 299-318.
- Oliver N., Rosario B., Pentland A. (2000). A bayesian computer vision system for modeling human interactions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, n° 8, p. 831-843.
- Orbanz P., Teh Y. W. (2011). Modern Bayesian nonparametrics. Tutorial presented in Neural Information Processing Systems.
- Pitman J. (2002). Poisson-dirichlet and gem invariant distributions for split-and-merge transformations of an interval partition. *Combinatorics, Probability and Computing*, vol. 11, p. 501–514.
- Rao C., Yilmaz A., Shah M. (2002). View-invariant representation and recognition of actions. *International Journal of Computer Vision*, vol. 50, n° 2, p. 203-226.

- Rodríguez A., Dunson D. B., Gelfand A. E. (2008). The nested Dirichlet process. *Journal of the American Statistical Association*, vol. 103, n° 483, p. 1131-1154.
- Schaefer A. T., Claridge-Chang A. (2012). The surveillance state of behavioral automation. *Current opinion in neurobiology*, vol. 22, n° 1, p. 170-176.
- Sethuraman J. (1994). A constructive definition of Dirichlet priors. *Statistica Sinica*, vol. 4, p. 639-650.
- Stauffer C., Grimson W. (2000). Learning patterns of activity using real-time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, n° 8, p. 747-757.
- Sudderth E. B., Torralba A., Freeman W. T., Willsky A. S. (2005). Learning hierarchical models of scenes, objects, and parts. In *Proceedings of the 10th IEEE International Conference on Computer Vision, 2005*, p. 1331-1338. Washington, DC, USA, IEEE Computer Society.
- Teh Y. W. (2007). Dirichlet processes: Tutorial and practical course.
- Teh Y. W., Jordan M. I. (2010). Hierarchical bayesian nonparametric models with applications. *Bayesian nonparametrics*, p. 158-207.
- Teh Y. W., Jordan M. I., Beal M. J., Blei D. M. (2006). Hierarchical dirichlet processes. *Journal of the American Statistical Association*, vol. 101, n° 476.
- Walker S. G., Damien P., Laud P. W., Smith A. F. M. (1999). Bayesian nonparametric inference for random distributions and related functions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 61, n° 3, p. 485-527.
- Wang X., Grimson W. E. L., Westin C.-F. (2011). Tractography segmentation using a hierarchical Dirichlet processes mixture model. *NeuroImage*, vol. 54, n° 1, p. 290-302.
- Wang X., Ma X., Grimson W. (2009). Unsupervised activity perception in crowded and complicated scenes using hierarchical bayesian models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, n° 3, p. 539-555.
- Wang X., Tieu K., Grimson E. (2006). Learning semantic scene models by trajectory analysis. In A. Leonardis, H. Bischof, A. Pinz (Eds.), *European Conference on Computer Vision, ECCV 2006*, vol. 3953, p. 110-123. Springer Berlin Heidelberg.
- Wulsin D., Litt B., Jensen S. T. (2012). A hierarchical dirichlet process model with multiple levels of clustering for human eeg seizure modeling. In J. Langford, J. Pineau (Eds.), *Proceedings of the 29th International Conference on Machine Learning, 2012*, p. 57-64. New York, NY, USA, ACM.
- Yang J., Li Y., Wang K., Wu Y., Altieri G., Scalia M. (2011). Mixed signature: an invariant descriptor for 3D motion trajectory perception and recognition. *Mathematical Problems in Engineering*, vol. 2012.
- Zelnik-Manor L., Irani M. (2001). Event-based analysis of video. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2001*, vol. 2, p. II-123-II-130 vol.2.
- Zhong H., Shi J., Visontai M. (2004). Detecting unusual activity in video. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004*, vol. 2, p. II-819-II-826 Vol.2.