

---

# Estimation de densités non paramétriques et multimodales par permutation de sous-particules

## Application au suivi d'un ou de plusieurs objets synthétiques articulés

Séverine Dubuisson\* — Christophe Gonzales\*

\* Laboratoire LIP6 (CNRS - UPMC) — 4 place Jussieu, 75005 Paris  
severine.dubuisson@lip6.fr, christophe.gonzales@lip6.fr

---

*RÉSUMÉ.* Dans cet article, nous proposons une approche originale d'estimation séquentielle de densités non paramétriques définies dans des espaces de grande dimension, dans le cadre méthodologique du filtrage particulaire. En exploitant les indépendances conditionnelles de l'espace d'état, nous proposons de permuter des sous-ensembles indépendants de particules de manière à générer un nouvel ensemble échantillonnant mieux cet espace. Nous intégrons cette approche dans deux versions classiques du filtre particulaire : celui avec échantillonnage partitionné et celui à recuit simulé de manière à prouver son efficacité. Nous nous comparons aux approches classiques dans le cadre de l'estimation des densités d'objets synthétiques articulés. Nous montrons que notre approche diminue à la fois les erreurs d'estimation et les temps de traitement.

*ABSTRACT.* In this paper, we propose an original approach for sequential non-parametric density estimation defined in high-dimensional state spaces using the particle filtering framework. By exploiting conditional independences in the state space, we propose to swap independent sub-particle sets to generate new sets that better sample this space. We integrate this approach into two versions of particle filter, i.e., partition sampling and annealed particle filter, to prove its efficiency. We compare it to classical approaches on synthetic articulated object density estimation problems, and show our approach reduces both estimation errors and computation times.

*MOTS-CLÉS :* approximation de densité, filtrage particulaire, réseaux bayésiens.

*KEYWORDS:* density approximation, particle filtering, Bayesian networks.

## 1. Introduction

En intelligence artificielle comme en vision, il est très important aujourd'hui d'être capable de traiter des problèmes définis sur des espaces d'état et d'observation de grande dimension. En vision notamment, l'augmentation de la qualité des données à traiter, et donc de la quantité d'information qu'on en extrait, et leurs multiples provenances (multi-caméra, multi-modalité, *etc.*) induisent ce type de situation. Dans le cadre de cet article, nous nous intéressons à l'estimation séquentielle de densités non paramétriques, multimodales et de grande dimension qui évoluent au cours du temps, et nous avons choisi de nous positionner dans le cadre méthodologique des méthodes séquentielles de Monte Carlo (filtre particulaire). Le choix de ce cadre méthodologique vient du fait que, à la différence d'autres approches plus classiques d'estimations séquentielles de densité dans un cadre bayésien, telles que le filtre de Kalman et ses dérivées (Gordon *et al.*, 1993), nous nous situons dans un contexte dans lequel nous ne pouvons faire aucune hypothèse, ni sur les densités que nous cherchons à estimer, ni sur la linéarité des fonctions que nous utilisons pour modéliser notre problème. Le filtre particulaire, de par ses propriétés d'échantillonnage de l'espace d'état (voir la section 2 à ce sujet), permet de maintenir un ensemble d'hypothèses (particules) de manière à estimer plus efficacement les densités. Il a toutefois été montré dans (MacCormick *et al.*, 2000) que le nombre de particules nécessaires pour une bonne estimation de la densité augmente de manière exponentielle avec la dimension de l'espace d'état, ce qui rend ces méthodes séquentielles inutilisables en temps réel.

Dans cet article, nous proposons d'exploiter les indépendances conditionnelles dans l'espace d'état en transformant, à l'aide d'une procédure originale de permutation sur des sous-espaces d'état des ensembles de particules pondérées, en de nouveaux ensembles qui échantillonnent mieux la densité, et en particulier ses modes. Cette approche nous permet ainsi de gérer les grands espaces d'état, d'améliorer la qualité d'estimation tout en réduisant le nombre de particules nécessaires et donc les temps de calcul.

Cet article est organisé de la manière suivante. La section 2 donne des rappels sur le filtrage particulaire, les notations qui seront utilisées, ainsi qu'un bref état de l'art sur la prise en compte des grands espaces d'état dans ce type d'approches. La section 3 détaille l'approche que nous proposons consistant à permuter des sous-échantillons indépendants. La section 4 donne les résultats expérimentaux en termes de qualité d'estimation et de temps de calcul dans le cadre d'estimation de densités d'objets articulés synthétiques. Nous étendons notre étude en particulier au cas d'estimations simultanées de densités. Enfin, nous concluons et donnons quelques perspectives dans la section 5.

## 2. Filtrage particulaire (PF)

### 2.1. Contexte théorique

Dans le contexte méthodologique du filtre particulaire (Gordon *et al.*, 1993), on cherche à estimer une séquence d'états  $\{\mathbf{x}_t\}_{t=1,\dots,T}$  dont l'évolution est donnée par l'équation :

$$\mathbf{x}_{t+1} = \mathbf{f}_{t+1}(\mathbf{x}_t, \mathbf{n}_{t+1}^x) \quad [1]$$

à partir d'un ensemble d'observations  $\{\mathbf{y}_t\}_{t=1,\dots,T}$ , reliées à l'état par l'équation :

$$\mathbf{y}_{t+1} = \mathbf{h}_{t+1}(\mathbf{x}_{t+1}, \mathbf{n}_{t+1}^y) \quad [2]$$

Classiquement,  $\mathbf{f}_{t+1}$  et  $\mathbf{h}_{t+1}$  sont des fonctions non linéaires variant dans le temps, et  $\mathbf{n}_{t+1}^x$  et  $\mathbf{n}_{t+1}^y$  des séquences de bruits i.i.d. Ces équations amènent naturellement à représenter ce type de problème à l'aide de la chaîne de Markov de la figure 1. Si l'on considère les équations ci-dessus d'un point de vue probabiliste, l'estimation que l'on cherche revient à calculer  $p(\mathbf{x}_{t+1}|\mathbf{y}_{1:t})$ , où  $\mathbf{y}_{1:t}$  représente l'ensemble des observations des pas de temps 1 à  $t$ . on peut alors calculer cette densité en deux étapes. D'abord, une étape de prédiction de la densité :

$$p(\mathbf{x}_{t+1}|\mathbf{y}_{1:t}) = \int_{\mathbf{x}_t} p(\mathbf{x}_{t+1}|\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{y}_{1:t})d\mathbf{x}_t \quad [3]$$

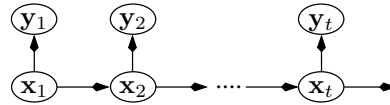
avec  $p(\mathbf{x}_{t+1}|\mathbf{x}_t)$  la densité *a priori* correspondant à la fonction de transition  $\mathbf{f}_{t+1}$ , puis une étape de correction

$$p(\mathbf{x}_{t+1}|\mathbf{y}_{1:t+1}) \propto p(\mathbf{y}_{t+1}|\mathbf{x}_{t+1})p(\mathbf{x}_{t+1}|\mathbf{y}_{1:t}) \quad [4]$$

avec  $p(\mathbf{y}_{t+1}|\mathbf{x}_{t+1})$  la vraisemblance, correspondant à la fonction d'observation  $\mathbf{h}_{t+1}$ .

Le filtrage particulaire (cf. algorithme 1) a pour but d'approcher ces distributions par un ensemble de  $N$  échantillons pondérés  $\{\mathbf{x}_{t+1}^{(i)}, w_{t+1}^{(i)}\}$ , ou *particules*, qui correspondent à des réalisations possibles de l'état. Là encore, il se divise en deux étapes :

1) une étape de prédiction (utilisant les estimations et observations passées), qui consiste à propager l'ensemble de particules  $\{\mathbf{x}_t^{(i)}, w_t^{(i)}\}$  selon une fonction de proposition  $q(\mathbf{x}_{t+1}|\mathbf{x}_{0:t}^{(i)}, \mathbf{y}_{t+1})$ , qui peut éventuellement différer de la densité *a priori* de transition  $p(\mathbf{x}_{t+1}|\mathbf{x}_t^{(i)})$ . Cette étape est suivie par :



**Figure 1.** Chaîne de Markov pour l'estimation d'états

**Input:** ensemble de particules  $\{\mathbf{x}_t^{(i)}, w_t^{(i)}\}$ , image  $\mathcal{I}$   
**Output:** ensemble de particules  $\{\mathbf{x}_{t+1}^{(i)}, w_{t+1}^{(i)}\}$   
 $\{\hat{\mathbf{x}}_{t+1}^{(i)}\} \leftarrow$  propagation  $(\{\mathbf{x}_t^{(i)}, w_t^{(i)}\})$   
 $\{\hat{w}_{t+1}^{(i)}\} \leftarrow$  correction  $(\{\hat{\mathbf{x}}_{t+1}^{(i)}, w_t^{(i)}\}, \mathcal{I})$   
 $\{\mathbf{x}_{t+1}^{(i)}, w_{t+1}^{(i)}\} \leftarrow$  rééchantillonnage  $(\{\hat{\mathbf{x}}_{t+1}^{(i)}, \hat{w}_{t+1}^{(i)}\})$   
**return**  $\{\mathbf{x}_{t+1}^{(i)}, w_{t+1}^{(i)}\}$

**Algorithme 1.** Filtre particulaire PF

2) une correction de cette prédiction (en utilisant l'observation courante  $\mathbf{y}_{t+1}$ ), consistant à pondérer les particules à l'aide d'une fonction de vraisemblance, de manière à avoir  $w_{t+1}^{(i)} \propto w_t^{(i)} p(\mathbf{y}_{t+1} | \mathbf{x}_{t+1}^{(i)}) \frac{p(\mathbf{x}_{t+1}^{(i)} | \mathbf{x}_t^{(i)})}{q(\mathbf{x}_{t+1}^{(i)} | \mathbf{x}_t^{(i)}, \mathbf{y}_{t+1})}$ , avec  $\sum_{i=1}^N w_{t+1}^{(i)} = 1$ .

Les particules peuvent alors être rééchantillonnées. Il existe plusieurs algorithmes pour réaliser cette tâche (Douc *et al.*, 2005), mais le principe est toujours le même : les particules de plus forts poids sont dupliquées, et celles avec les plus faibles sont éliminées.

L'estimation de la densité *a posteriori* est alors donnée par  $\sum_{i=1}^N w_{t+1}^{(i)} \delta_{\mathbf{x}_{t+1}^{(i)}}(\mathbf{x}_{t+1})$ , où  $\delta_{\mathbf{x}_{t+1}^{(i)}}$  sont des masses de Dirac centrées sur les particules  $\mathbf{x}_{t+1}^{(i)}$ .

## 2.2. Gestion des grands espaces d'état

Afin d'obtenir des traitements en un temps raisonnable, la manipulation des distributions de probabilité ci-dessus nécessite des espaces d'état et d'observation de petites dimensions. Lorsque ce n'est pas le cas, il existe essentiellement deux manières pour rendre le processus viable : soit en simplifiant les espaces à parcourir par le biais de recherches locales "astucieuses", soit en faisant des approximations des calculs ou en décomposant ces espaces en produits cartésiens de sous-espaces de tailles raisonnables et en exploitant les indépendances conditionnelles probabilistes entre ces derniers.

Pour la première famille d'approches, et parmi celles qui ont été intégrées dans un filtre particulaire, on citera en particulier les méthodes d'optimisation. Leur principe est de systématiquement proposer une méthode d'exploration de l'espace d'état de manière à trouver les modes de la distribution à estimer, selon une fonction objectif à minimiser ou maximiser (en général le contenu d'une image centrée sur la position évaluée est comparé à celui d'une cible, et on cherche à minimiser la distance entre les modèles qui représentent ces contenus). La différence d'une méthode à l'autre résulte essentiellement du choix de la fonction d'exploration utilisée, dans laquelle plus ou moins d'intelligence est incorporée de manière à éviter de tomber dans des maxima locaux. La méthode de Kanade-Lucas-Tomasi (KLT) (Lucas *et al.*, 1981) utilise une simple méthode de localisation *via* le gradient de l'image avec, comme fonction ob-

jectif à maximiser, une mesure de corrélation. Le *Mean-Shift* (Comaniciu *et al.*, 2003) est en essence similaire, sauf que la fonction objectif est cette fois une mesure de distance de Bhattacharyya. Dans (Bray *et al.*, 2005), une nouvelle technique d'optimisation est proposée, plus efficace qu'une simple descente de gradient car elle intègre des contraintes liées au contexte applicatif. Toute une famille d'approches utilisent également des heuristiques. Une métaheuristique fondée sur une population est exploitée dans un schéma de *path relinking* (Sánchez *et al.*, 2005) et est utilisée pour le rééchantillonnage des particules afin d'éviter de passer à côté des modes de la densité à estimer. De même, la méthode dite de *scatter search* (Pantrigo *et al.*, 2008) est basée sur la construction généralisée de chemins dans l'espace euclidien. Les algorithmes génétiques (Han *et al.*, 2005) utilisent des mécanismes fondés sur la génétique naturelle et l'évolution darwinienne. L'algorithme bio-inspiré IWO (Ahmadi *et al.*, 2012) imite le comportement d'une colonie de fourmis via des phénomènes de croissances, compétitions, *etc.* Parmi les autres approches fondées sur le concept de population, on peut citer les algorithmes évolutionnaires, qui sont capables de détecter des solutions multiples et sont donc bien adaptées au cas multimodal. Enfin, les algorithmes PSO (*Particle swarm optimization*) (Zhao *et al.*, 2010; John *et al.*, 2010) sont probablement les plus utilisés actuellement. Ils cherchent à simuler un comportement social observé chez les animaux. Dans ces approches, les particules subissent simultanément des accélérations ou décélérations dépendant de leur position précédente mais aussi de leur position par rapport à la population. Dans le cadre de cet article, nous nous focalisons sur le filtre particulaire à recuit simulé (*Annealed Particle Filter – APF*) (Deutscher *et al.*, 2005), qui est probablement le plus utilisé et qui consiste à ajouter à l'étape de rééchantillonnage des itérations (ou couches) de recuit simulé, de manière à mieux diffuser les particules dans l'espace d'état (cf. algorithme 3). Le principe est de diviser l'étape de correction en un ensemble de couches, dans lesquelles on diffuse les particules puis on les corrige. La diffusion des particules est de plus en plus concentrée, et la vraisemblance utilisée pour la correction de moins en moins lissée (un peu comme les principes appliqués en analyse multiéchelle).

Le problème de la première famille d'approches est que l'on sort très souvent du cadre théorique du filtrage particulaire. En effet, le fait de déplacer les particules indépendamment les unes des autres dans l'espace de recherche fait qu'on change la distribution à chaque pas de temps. Seul l'APF permet de rester dans ce cadre, du

**Input:** particules  $\{\mathbf{x}_t^{(i)}, w_t^{(i)}\}$  de la couche  $l$ , image  $\mathcal{I}$   
**Output:** particules  $\{\mathbf{x}_t^{(i)}, w_t^{(i)}\}$  de la couche  $l + 1$   
 $\{\tilde{w}_t^{(i)}\} \leftarrow$  changement d'échelle ( $\{w_t^{(i)}\}$ )  
 $\{\bar{\mathbf{x}}_t^{(i)}, \bar{w}_t^{(i)}\} \leftarrow$  rééchantillonnage ( $\{\mathbf{x}_t^{(i)}, \tilde{w}_t^{(i)}\}$ )  
 $\{\hat{\mathbf{x}}_t^{(i)}\} \leftarrow$  propagation ( $\{\bar{\mathbf{x}}_t^{(i)}, \bar{w}_t^{(i)}\}$ )  
 $\{\hat{w}_t^{(i)}\} \leftarrow$  correction ( $\{\hat{\mathbf{x}}_t^{(i)}, \bar{w}_t^{(i)}\}, \mathcal{I}$ )  
**return**  $\{\hat{\mathbf{x}}_t^{(i)}, \hat{w}_t^{(i)}\}$

**Algorithme 2.** Annealing

**Input:**  $\{\mathbf{x}_t^{(i)}, w_t^{(i)}\}$ , image  $\mathcal{I}$ , nombre de couches  $L$   
**Output:** particules  $\{\mathbf{x}_{t+1}^{(i)}, w_{t+1}^{(i)}\}$  au temps  $t + 1$   
 $\{\mathbf{x}_{t+1,0}^{(i)}, w_{t+1,0}^{(i)}\} \leftarrow \text{PF}(\{\mathbf{x}_t^{(i)}, w_t^{(i)}\})$   
**for**  $l = 1$  **to**  $L$  **do**  
     $\{\mathbf{x}_{t+1,l}^{(i)}, w_{t+1,l}^{(i)}\} \leftarrow \text{Annealing}(\{\mathbf{x}_{t+1,l-1}^{(i)}, w_{t+1,l-1}^{(i)}\})$   
**return**  $\{\mathbf{x}_{t+1,L}^{(i)}, w_{t+1,L}^{(i)}\}$

**Algorithme 3.** Le filtre particulaire à recuit simulé APF

fait des rééchantillonnages qu'il effectue après chaque couche de recuit simulé. De plus, les métaheuristiques sur lesquelles sont basées la majeure partie des méthodes d'optimisation décrites plus haut ne possèdent pas de propriétés mathématiques qui permettent de disposer d'un cadre rigoureux nous assurant soit la convergence, soit l'exactitude des résultats.

La seconde famille d'approches consiste à décomposer les calculs en sous-calculs dits "réalisables". Nous pouvons citer comme exemple significatif le *Partitioned Sampling* (PS) (MacCormick *et al.*, 1999), qui exploite la décomposition de la densité de vraisemblance ainsi que de la dynamique dans des espaces de dimension réduite. Ainsi, les espaces d'état  $\mathcal{X}$  et d'observation  $\mathcal{Y}$  sont décomposés de la manière suivante :  $\mathcal{X} = \mathcal{X}^1 \times \dots \times \mathcal{X}^P$  et  $\mathcal{Y} = \mathcal{Y}^1 \times \dots \times \mathcal{Y}^P$ . Par exemple, afin de suivre les mouvements d'une personne,  $\mathcal{X}$  se décompose naturellement comme  $\mathcal{X}^{\text{buste}} \times \mathcal{X}^{\text{bras gauche}} \times \mathcal{X}^{\text{bras droit}}$ . L'exploitation d'indépendances conditionnelles entre ces sous-espaces amène à appliquer l'algorithme PF non plus en une seule fois sur  $(\mathcal{X}, \mathcal{Y})$  mais séquentiellement sur chacun des  $(\mathcal{X}^i, \mathcal{Y}^i)$ , ce qui accélère considérablement les temps de calcul. En effet, si l'on connaît la position du buste de la personne à suivre, on peut raisonnablement penser qu'il y a indépendance entre la position du bras gauche et celle du bras droit. Cela suggère donc de décomposer la distribution jointe sur  $(\mathcal{X}, \mathcal{Y})$  et sur l'ensemble des pas de temps  $t$ , comme indiqué dans le réseau bayésien dynamique (DBN) de la figure 2, où le buste correspond à la variable  $\mathbf{x}_t^1$ , le bras gauche à  $\mathbf{x}_t^2$  et le bras droit à  $\mathbf{x}_t^3$  :

### Définition 1 (réseau bayésien dynamique (DBN) (Murphy, 2002))

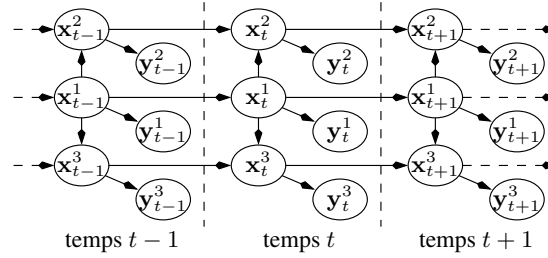
Un réseau bayésien est un couple  $(G, \mathbf{P})$ , où  $G = (\mathbf{V}, \mathbf{A})$  est un graphe orienté sans circuit. Chaque nœud  $X \in \mathbf{V}$  correspond à une variable aléatoire<sup>1</sup>.  $\mathbf{P} = \{p(X|\mathbf{Pa}(X)) : X \in \mathbf{V}\}$ , où  $\mathbf{Pa}(X)$  est l'ensemble des parents de  $X$  dans  $G$ , est un ensemble de probabilités (ou densités) conditionnelles, et la probabilité jointe  $p(\mathbf{V})$  est égale à  $p(\mathbf{V}) = \prod_{X \in \mathbf{V}} p(X|\mathbf{Pa}(X))$ .

1. Par abus de notation, puisqu'il y a bijection entre les nœuds de  $\mathbf{V}$  et les variables aléatoires correspondantes, nous utiliserons  $X \in \mathbf{V}$  pour représenter indifféremment le nœud  $X$  ou sa variable aléatoire.

**Input:** particules  $\{\mathbf{x}_t^{(i)}, w_t^{(i)}\}$  au temps  $t$ , image  $\mathcal{I}$   
**Output:** particules  $\{\mathbf{x}_{t+1}^{(i)}, w_{t+1}^{(i)}\}$  au temps  $t + 1$   
 $\{\mathbf{x}^{(i)}, w^{(i)}\} \leftarrow \{\mathbf{x}_t^{(i)}, w_t^{(i)}\}$   
**for**  $j = 1$  **to**  $P$  **do**  
      $\{\hat{\mathbf{x}}^{(i),j}\} \leftarrow \text{propagation}(\{\mathbf{x}^{(i),j}\})$   
      $\{\hat{w}^{(i)}\} \leftarrow \text{correction}(\{\{\hat{\mathbf{x}}^{(i),j}, \mathbf{x}^{(i),-j}\}, w^{(i)}\}, \mathcal{I})$   
      $\{\mathbf{x}^{(i)}, w^{(i)}\} \leftarrow \text{rééchantillonnage}(\{\{\hat{\mathbf{x}}^{(i),j}, \mathbf{x}^{(i),-j}\}, \hat{w}^{(i)}\})$   
**return**  $\{\mathbf{x}^{(i)}, w^{(i)}\}$

**Algorithme 4.** Partitioned Sampling PS.

Un réseau bayésien dynamique est une extension d'un réseau bayésien incorporant une notion de "temporalité" : les nœuds de  $\mathbf{V}$  sont alors indexés par le temps :  $\mathbf{V} = \cup_t \{\mathbf{x}_t, \mathbf{y}_t\}$  comme le montre la figure 2.



**Figure 2.** Un réseau bayésien dynamique (DBN)

Revenons à notre problème de suivi des mouvements d'une personne, dans lequel le buste correspond à la variable  $\mathbf{x}_t^1$ , le bras gauche à  $\mathbf{x}_t^2$  et le bras droit à  $\mathbf{x}_t^3$ . Le modèle d'indépendance des DBN (la  $d$ -séparation, (Pearl, 1988), cf. ci-dessous) permet de démontrer qu'il y a indépendance entre la position du bras gauche et celle du bras droit conditionnellement à celle du buste. Cela justifie alors l'application séquentielle de PF sur chacun des sous-espaces  $(\mathcal{X}^i, \mathcal{Y}^i)$ ,  $i = 1, \dots, P$ . PS est décrit plus formellement dans l'algorithme 4, où la notation  $(\hat{\mathbf{x}}_t^{(i),j}, \mathbf{x}_t^{(i),-j})$  représente le fait que la  $i^{\text{ème}}$  particule a le même état que  $\hat{\mathbf{x}}_t^{(i)}$  sur la partie  $j$ , c'est-à-dire sur  $\mathcal{X}^j$ , et le même état que  $\mathbf{x}_t^{(i)}$  sur les autres parties. Le problème est qu'en multipliant les sous-espaces de traitement, on multiplie également le nombre d'étapes de rééchantillonnage, ce qui bruite énormément le processus d'estimation et dégrade les performances au cours du temps. Une idée similaire fondée sur les DBNs est exploitée dans (Kanazawa *et al.*, 1995) : la fonction de proposition  $q$  est décomposée comme le produit des densités conditionnelles en chaque nœud du réseau et PF est appliqué séquentiellement sur ces nœuds selon un ordre topologique dans le DBN. Ces deux approches sont combinées dans (Rose *et al.*, 2008), pour définir l'algorithme du filtre particulaire de manière à entièrement l'intégrer dans un DBN, ce qui peut être vu comme une généralisation de PS. D'autres méthodes fondées sur les réseaux bayésiens ont également

été développées pour gérer les espaces de grandes dimensions. Par exemple, la technique non paramétrique de propagation de croyance (*Belief Propagation*) (Sudderth *et al.*, 2010), ou encore le PF Rao-Blackwellisé pour DBN (RBPF) (Doucet *et al.*, 2000) qui décompose l'espace d'état en deux parties selon l'hypothèse forte que la distribution conditionnelle de la seconde partie peut être estimée par des méthodes classiques (filtre de Kalman par exemple). L'approche décrite dans (Besada-Portas *et al.*, 2009) est probablement la plus proche de la notre. On y décrit un algorithme parallélisé de PF qui utilise la même décomposition de la probabilité jointe que dans les DBN afin de réduire le nombre de particules. L'espace d'état est divisé en sous-espaces dans lesquels les particules sont générées de manière indépendante à l'aide de différentes densités de proposition  $q$ . Cette approche offre une manière très flexible de choisir  $q$  pour échantillonner chaque sous-espace. En revanche, cela demande au DBN d'avoir une structure d'indépendance particulière, ce qui limite la généralisation de l'algorithme. Dans cet article, nous adressons des problèmes plus généraux, ce qui est décrit dans la section suivante.

### 3. Filtrage particulière avec permutation de sous-particules

Comme nous l'avons vu, l'idée clé de PS et de ses variations consiste à exploiter les indépendances probabilistes encodées par les DBN afin de ne réaliser les calculs du filtre particulière que dans des sous-espaces  $(\mathcal{X}^i, \mathcal{Y}^i)$  de petite dimension. Ces espaces étant "petits", les distributions de probabilités conditionnelles à estimer ont également beaucoup moins de paramètres que celle définie sur  $(\mathcal{X}, \mathcal{Y})$ , ce qui réduit exponentiellement le nombre de particules nécessaires à leur estimation et accélère donc d'autant les calculs. Cependant, PS n'exploite pas optimalement les indépendances du DBN. En effet, celles-ci sont décrites par la propriété dite de  $d$ -séparation définie de la manière suivante :

**Définition 2 ( $d$ -séparation (Pearl, 1988))** Deux nœuds (variables aléatoires)  $\mathbf{x}_t^i$  et  $\mathbf{x}_s^j$  d'un DBN sont dépendants conditionnellement à un ensemble de nœuds  $\mathbf{Z}$  si et seulement s'il existe une chaîne, c'est-à-dire un chemin non orienté,  $\{\mathbf{c}_1 = \mathbf{x}_t^i, \dots, \mathbf{c}_n = \mathbf{x}_s^j\}$  reliant  $\mathbf{x}_t^i$  à  $\mathbf{x}_s^j$  dans le DBN, et satisfaisant les deux propriétés suivantes :

- 1) pour tout nœud  $\mathbf{c}_k$  de la chaîne dont les arcs adjacents sont de la forme  $\mathbf{c}_{k-1} \rightarrow \mathbf{c}_k \leftarrow \mathbf{c}_{k+1}$ , le nœud  $\mathbf{c}_k$  ou un de ses descendant appartient à  $\mathbf{Z}$  ;
- 2) aucun des autres nœuds  $\mathbf{c}_k$  n'appartient à  $\mathbf{Z}$ .

Une telle chaîne est dite active (sinon elle est bloquée). S'il existe une chaîne active reliant deux nœuds, ceux-ci sont probabilistiquement dépendants et sont dits  $d$ -connectés, sinon ils sont indépendants conditionnellement à  $\mathbf{Z}$  et sont dits  $d$ -séparés.

repreons le DBN de la figure 2. Par  $d$ -séparation,  $\mathbf{x}_t^2$  et  $\mathbf{x}_t^3$  sont indépendants conditionnellement à  $(\mathbf{x}_t^1, \mathbf{x}_{t-1}^2, \mathbf{x}_{t-1}^3)$ , ce qui implique que, dans PS, les parties 2 et



3 peuvent être propagées/corrigées en parallèle plutôt que séquentiellement tout en garantissant que la distribution estimée est correcte. Le premier avantage de procéder ainsi est que l'on n'a besoin que d'un seul rééchantillonnage au lieu de 2 pour le filtrage des parties 2 et 3. Le deuxième avantage, et c'est l'objet principal de cet article, est que l'on va pouvoir permuter, parmi certains ensembles de particules, les sous-états des parties 2 et 3 de manière à se concentrer sur les modes de la distribution à estimer sans toutefois modifier la distribution estimée elle-même. Cette idée a été initiée dans (Gonzales *et al.*, 2011) et nous la formalisons complètement ici.

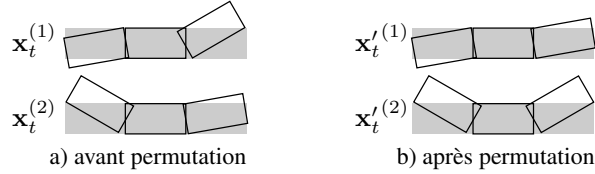
Pour cela, nous devons considérer, non plus les particules  $\{\mathbf{x}_t^{(i)}, w_t^{(i)}\}$  au temps  $t$ , mais leurs trajectoires totales, soit  $\{\mathbf{x}_{1:t}^{(i)}, w_{1:t}^{(i)}\}$ . À un facteur de normalisation près, celles-ci échantillonnent la distribution  $p(\mathbf{x}_{1:t}, \mathbf{y}_{1:t})$ . De la même manière que les états  $\mathbf{x}_t$  se décomposent en variables  $\mathbf{x}_t^j$ , les poids  $w_t$  se décomposent en  $w_t^j = p(\mathbf{y}_t^j | \mathbf{x}_t^j)$ , c'est-à-dire la vraisemblance de la partie  $j$ . Un poids  $w_t$  est alors défini par  $w_t \propto \prod_{j=1}^P w_t^j$ . Alors,  $\{\mathbf{x}_{1:t}^{(i,j)}, w_{1:t}^{(i,j)}\}$  échantillonne la distribution  $p(\mathbf{x}_{1:t}^j, \mathbf{y}_{1:t}^j | \mathbf{Pa}(\mathbf{x}_{1:t}^j))$ , où  $\mathbf{Pa}(\mathbf{x}_{1:t}^j)$  représente  $\bigcup_{s=1}^t \{\text{parents des nœuds } \mathbf{x}_s^j \text{ dans le DBN}\}$ . Et :

$$p(\mathbf{x}_{1:t}, \mathbf{y}_{1:t}) = \prod_{j=1}^P p(\mathbf{x}_{1:t}^j, \mathbf{y}_{1:t}^j | \mathbf{Pa}(\mathbf{x}_{1:t}^j)).$$

Par exemple, sur le DBN de la figure 2,  $\{\mathbf{x}_{1:t}^{(i,2)}, w_{1:t}^{(i,2)}\}$ , c'est-à-dire l'ensemble des parties 2 des particules sur toutes les tranches de temps, échantillonne  $p(\mathbf{x}_{1:t}^2, \mathbf{y}_{1:t}^2 | \mathbf{x}_{1:t}^1)$ , la distribution de toutes les variables sur la ligne du haut conditionnellement à celles du milieu. De même,  $\{\mathbf{x}_{1:t}^{(i,1)}, w_{1:t}^{(i,1)}\}$  et  $\{\mathbf{x}_{1:t}^{(i,3)}, w_{1:t}^{(i,3)}\}$  échantillonnent respectivement  $p(\mathbf{x}_{1:t}^1, \mathbf{y}_{1:t}^1)$  et  $p(\mathbf{x}_{1:t}^3, \mathbf{y}_{1:t}^3 | \mathbf{x}_{1:t}^1)$ , et la loi jointe se décompose en produit des 3 distributions :

$$p(\mathbf{x}_{1:t}^1, \mathbf{y}_{1:t}^1) \times p(\mathbf{x}_{1:t}^2, \mathbf{y}_{1:t}^2 | \mathbf{x}_{1:t}^1) \times p(\mathbf{x}_{1:t}^3, \mathbf{y}_{1:t}^3 | \mathbf{x}_{1:t}^1).$$

Or, par  $d$ -séparation, l'ensemble  $\{\mathbf{x}_{1:t}^2, \mathbf{y}_{1:t}^2\}$  est indépendant de  $\{\mathbf{x}_{1:t}^3, \mathbf{y}_{1:t}^3\}$  conditionnellement à  $\{\mathbf{x}_{1:t}^1\}$ . Donc, si l'on permute deux "sous-particules"  $\{\mathbf{x}_{1:t}^{(i,3)}, w_{1:t}^{(i,3)}\}$  et  $\{\mathbf{x}_{1:t}^{(j,3)}, w_{1:t}^{(j,3)}\}$  telles que  $\{\mathbf{x}_{1:t}^{(i,1)}\} = \{\mathbf{x}_{1:t}^{(j,1)}\}$ , cela n'aura d'impact ni sur l'estimation de  $p(\mathbf{x}_{1:t}^2, \mathbf{y}_{1:t}^2 | \mathbf{x}_{1:t}^1)$  (par indépendance) ni sur celle de  $p(\mathbf{x}_{1:t}^3, \mathbf{y}_{1:t}^3 | \mathbf{x}_{1:t}^1)$  (puisque permuter des éléments dans un échantillon n'affecte en rien l'estimation qu'il fournit). *A fortiori*, la distribution jointe estimée est inchangée. Cependant, cette opération permet de se focaliser sur les modes de cette distribution. En effet, considérons la figure 3, où 2 particules sont représentées,  $\mathbf{x}_t^1, \mathbf{x}_t^2, \mathbf{x}_t^3$  étant respectivement les parties centrale, gauche et droite de l'objet et les parties grisées représentant l'état réel de l'objet. Ces particules ont la même valeur pour  $\mathbf{x}_t^1$ , on peut donc permuter leurs valeurs de  $\mathbf{x}_t^3$ . Notons qu'avant permutation, la partie droite (resp. gauche) de la 1<sup>re</sup> (resp. 2<sup>e</sup>) particule était éloignée de l'état réel et induisait donc, globalement, un poids  $w_t$  peu élevé à la particule. Après permutation, la 1<sup>re</sup> particule est très proche de l'état réel (et a donc un fort poids) et la deuxième en est assez éloignée (et a donc un très faible poids). Après rééchantillonnage, cette dernière sera donc supprimée et il ne restera que la 1<sup>re</sup> particule, proche du mode de  $p$ .



**Figure 3.** L'opération de permutation

Plus formellement, soit  $\mathcal{G}$  un DBN de variables aléatoires  $(\mathbf{x}_{1:t}^i, \mathbf{y}_{1:t}^i)$ ,  $i = 1, \dots, P$ . Comme usuellement, nous supposons que  $\mathcal{G}$  est un 2TBN (Murphy, 2002), c'est-à-dire que :

- $\mathcal{G}$  ne contient que des arcs dont les extrémités appartiennent soit à la même tranche de temps soit à 2 tranches consécutives ;
- le réseau est identique sur chaque tranche de temps.

De plus, afin de simplifier la présentation de l'opération de permutation sans toutefois vraiment restreindre son champ d'application, nous supposons qu'à l'intérieur d'une tranche de temps, le réseau est une forêt (chaque nœud a au plus un parent dans la tranche) et que les seuls arcs entre les tranches sont du type  $(\mathbf{x}_{t-1}^j, \mathbf{x}_t^j)$ . Le DBN de la figure 2 vérifie ces hypothèses. Pour tout  $j \in \{1, \dots, P\}$ , appelons  $\mathbf{Desc}(\{\mathbf{x}_t^j\})$  l'ensemble des descendants des variables de  $\{\mathbf{x}_t^j\}$  dans la tranche de temps  $t$ , et  $\mathbf{Desc}(\{\mathbf{x}_{1:t}^j\}) = \cup_{s=1}^t \mathbf{Desc}(\{\mathbf{x}_s^j\})$ . Par exemple, sur la figure 2,  $\mathbf{Desc}(\{\mathbf{x}_t^1\}) = \{\mathbf{x}_t^2, \mathbf{x}_t^3\}$  et  $\mathbf{Desc}(\{\mathbf{x}_{1:t}^1\}) = \{\mathbf{x}_{1:t}^2, \mathbf{x}_{1:t}^3\}$ . Soit  $\{P_1, \dots, P_K\}$  une partition de  $\{1, \dots, P\}$  telle que :

$$\forall i, \forall j \in P_i, \left( \exists \mathbf{x}_t^k \in \mathcal{G} \text{ tel que } \mathbf{x}_t^j \in \mathbf{Desc}(\{\mathbf{x}_t^k\}) \right) \Rightarrow k \in P_h \text{ avec } h < i.$$

Sur la figure 2,  $P_1 = \{\mathbf{x}_t^1\}$ ,  $P_2 = \{\mathbf{x}_t^2, \mathbf{x}_t^3\}$  est une telle partition. Intuitivement,  $P_1$  représente l'ensemble des nœuds sans parent dans la même tranche de temps,  $P_2$  est l'ensemble des enfants de  $P_1$ ,  $P_3$  l'ensemble des enfants de  $P_2$ , etc. Chaque  $P_i$  correspond à un ensemble de parties que PS devrait propager/corriger en parallèle au temps  $t$ . Pour un  $P_i$  donné, l'opération de permutation, que nous noterons  $\rightleftharpoons^{P_i}$  est définie par : pour tout  $j \in P_i$ , soit  $\sigma_j$  une permutation des index des particules telle que seules des sous-particules ayant une même valeur de  $\mathbf{Pa}(\mathbf{x}_{1:t}^j)$  peuvent être permutes, alors toutes les sous-particules appartenant à  $\{\mathbf{x}_{1:t}^j\} \cup_{k \in \mathbf{Desc}(\{\mathbf{x}_{1:t}^j\})} \{\mathbf{x}_{1:t}^k\}$  sont permutes selon  $\sigma_j$ .

**Proposition 1** En propageant / corrigeant / permutant (selon  $\rightleftharpoons^{P_i}$ ) séquentiellement les sous-particules de  $P_i$ ,  $i = 1, \dots, K$ , on échantillonne la distribution  $p(\mathbf{x}_{1:t} | \mathbf{y}_{1:t})$ .

La démonstration de cette proposition est fournie en appendice. Comme on travaille sur des espaces continus, la probabilité que 2 sous-particules quelconques aient

une même valeur de  $\mathbf{Pa}(\mathbf{x}_t^j)$  est extrêmement faible. En pratique, cela n'arrive que lorsqu'elles ont été engendrées par rééchantillonnage à partir d'une même particule. Dans ce cas, elles ont également une même valeur de  $\mathbf{Pa}(\mathbf{x}_{1:t}^j)$ . On s'autorise donc ici à permuter des particules dès lors qu'elles ont une même valeur de  $\mathbf{Pa}(\mathbf{x}_t^j)$ . L'avantage est que l'on n'a plus besoin de considérer les trajectoires des particules du temps 1 à  $t$ , mais seulement sur la tranche de temps  $t$ , ce qui conduit à l'algorithme 5. APF peut également être étendu afin d'intégrer l'opération de permutation. Pour cela, différentes possibilités existent, mais celle qui semble la plus efficace consiste à rajouter après chaque  $\rightleftharpoons^{P_i}$  de  $\overline{\overline{PS}}$  des couches d'annealing portant sur  $P_i$  suivies, chacune, par une permutation  $\rightleftharpoons^{P_i}$ . Cet algorithme est appelé  $\overline{\overline{APF}}$ .

**Input:**  $\{\mathbf{x}_t^{(i)}, w_t^{(i)}\}$ , partition  $\{P_1, \dots, P_K\}$ , image  $\mathcal{I}$   
**Output:** particules  $\{\mathbf{x}_{t+1}^{(i)}, w_{t+1}^{(i)}\}$  au temps  $t + 1$   
 $\{\mathbf{x}^{(i)}, w^{(i)}\} \leftarrow \{\mathbf{x}_t^{(i)}, w_t^{(i)}\}$   
**for**  $j = 1$  **to**  $K$  **do**  
     **for**  $k = 1$  **to**  $|P_j|$  **do**  
          $\{\mathbf{x}^{(i),k}\} \leftarrow$  propagation  $(\{\mathbf{x}^{(i),k}\})$   
          $\{w^{(i),k}\} \leftarrow$  correction  $(\{\mathbf{x}^{(i),k}\}, \mathcal{I})$   
          $\{\bar{\mathbf{x}}^{(i)}, \bar{w}^{(i)}\} \leftarrow \rightleftharpoons^{P_i} (\{\mathbf{x}^{(i)}, w^{(i)}\})$   
          $\{\mathbf{x}^{(i)}, w^{(i)}\} \leftarrow$  rééchantillonnage  $(\{\bar{\mathbf{x}}^{(i)}, \bar{w}^{(i)}\})$   
**return**  $\{\mathbf{x}^{(i)}, w^{(i)}\}$

**Algorithme 5.** Filtre avec permutation  $\overline{\overline{PS}}$

Évidemment, certaines permutations sont meilleures que d'autres pour se focaliser sur les modes de la distribution. Une “bonne” permutation peut être obtenue assez simplement : d’après l’argument ci-dessus, si, pour  $j \in P_i$ , deux particules ont une même valeur de  $\mathbf{Pa}(\mathbf{x}_t^j)$ , elles ont probablement été engendrées par rééchantillonnage à partir d'une même particule. Donc, pour tout  $k \in P_i$ , elles ont également une même valeur de  $\mathbf{Pa}(\mathbf{x}_t^k)$ . On peut donc partitionner l'ensemble  $N$  de particules en des sous-ensembles  $N_1, \dots, N_R$  tels que les particules d'un ensemble  $N_r$ ,  $r = 1, \dots, R$ , ont la même valeur de  $\mathbf{Pa}(\mathbf{x}_t^j)$  pour un  $j \in P_i$ . Pour chaque  $N_r$ , toutes les permutations possibles sont admissibles. Appelons  $\{r_1, \dots, r_s\}$  les éléments de  $N_r$ . Pour chaque  $j \in P_i$ , soit  $\sigma_j$  la permutation correspondant au tri des poids  $w_t^{(r_h),j}$ ,  $h = 1, \dots, s$ , par ordre décroissant. En appliquant  $\sigma_j$  pour tout  $j \in P_i$ , on obtient une opération de permutation  $\rightleftharpoons^{P_i}$  associant à la première particule l'ensemble meilleurs poids, à la seconde l'ensemble des seconds meilleurs poids, etc. Ainsi les premières particules sont-elles de poids très élevés et les dernières de poids très faibles (elles seront supprimées au prochain rééchantillonnage). La complexité en temps d'un tel algorithme pour l'ensemble des  $P_i$  est en  $O(PN(E + \log N))$ , où  $E$  représente la taille des variables  $\mathbf{x}_t^j$ . En effet, pour un  $P_i$  donné, en utilisant une table de hachage, la détermination des  $N_r$  peut être effectuée en  $O(N)$ . Pour chaque  $N_r$ , il y a  $|P_i|$  listes à trier, ce qui peut être réalisé, globalement en  $|P_i|N \log N$ . Enfin, l'application de la permutation modifie au plus  $P|\mathbf{x}_t^j|$  par particule et se fait donc, globalement, en  $O(NPE)$ .

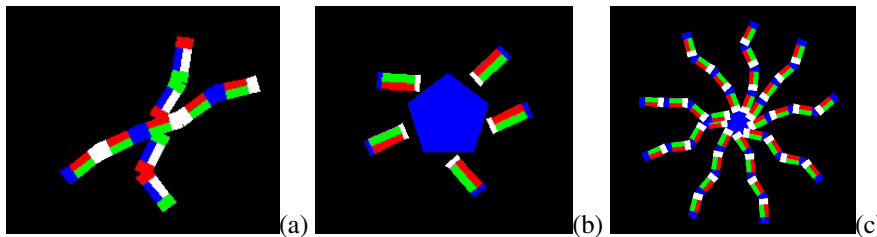
## 4. Résultats expérimentaux

Nous avons choisi de focaliser notre étude sur la qualité de l'estimation séquentielle d'une densité, sans nous contraindre à des problèmes purement liés à l'image dans un premier temps. Pour cette raison, nous testons cette approche et la comparons à d'autres sur des séquences vidéo synthétiques que nous avons nous-mêmes générées, et qui contiennent toutes 300 images. Si elles sont visuellement très simples, elles contiennent de réelles difficultés pour un estimateur séquentiel, comme nous le verrons par la suite.

### 4.1. Conditions expérimentales

#### 4.1.1. Espace d'état

Les objets dont nous cherchons à estimer la densité sont articulés. Ils se déplacent et se déforment de manière aléatoire : on les modélise par un ensemble  $P$  de parties (ou régions) représentées par des polygones de géométrie variable, dont les sommets sont étiquetés. Les objets ont une partie centrale  $P_1$ , dont part un ensemble de  $|P_i|$  "bras",  $i = 2, \dots, K$ , chacun de longueur  $K - 1$  (voir section 3). Le vecteur d'état contient les paramètres décrivant chaque partie, et est défini par  $\mathbf{x}_t = \{\mathbf{x}_t^1, \mathbf{x}_t^2, \dots, \mathbf{x}_t^P\}$ , avec  $\mathbf{x}_t^p = \{x_t^p, y_t^p, \theta_t^p\}$ , où  $(x_t^p, y_t^p)$  est le centre de la partie  $p$ , et  $\theta_t^p$  son orientation,  $p = 1, \dots, P$ . On a donc  $|\mathcal{X}| = 3P$  paramètres dans les vecteurs d'état. Une particule  $\mathbf{x}_t^{(j)} = \{\mathbf{x}_t^{(j),1}, \mathbf{x}_t^{(j),2}, \dots, \mathbf{x}_t^{(j),P}\}$ ,  $j = 1, \dots, N$ , est donc une configuration spatiale possible (c'est-à-dire une hypothèse dans l'espace d'état) de l'objet articulé. La figure 4.(a) montre un objet avec  $P = 9$ ,  $|P_i| = 4$  et  $K = 3$ , soit  $|\mathcal{X}| = 27$ . Sur la figure 4.(b), nous avons un objet avec  $P = 6$ ,  $|P_i| = 5$ , et  $K = 2$ , soit  $|\mathcal{X}| = 18$ . Enfin, sur la figure 4.(c), on a  $P = 37$ ,  $|P_i| = 9$  et  $K = 5$  soit  $|\mathcal{X}| = 111$ .



**Figure 4.** Exemples d'objets (zoom) dont on cherche à estimer la densité de probabilité. De gauche à droite :  $|\mathcal{X}| = 27, 18$  et  $111$

#### 4.1.2. Prédiction

Dans la première image, les particules sont générées uniformément autour de la position de l'objet. Pendant l'étape de prédiction, elles sont propagées selon une marche

aléatoire, centrée sur la position précédemment estimée, et dont la variance a été empiriquement choisie.

#### 4.1.3. Correction

Les poids des particules sont calculés à partir de l'observation (image) courante, et sont donnés par  $w_{t+1}^{(j)} = w_t^{(j)} p(\mathbf{y}_{t+1} | \mathbf{x}_{t+1}^{(j)}) \propto w_t^{(j)} e^{-\lambda d^2}$ , avec  $\lambda = 50$  et  $d$  la distance de Bhattacharyya entre les histogrammes cible (densité *a priori*) et référence (estimé à l'instant d'avant) de ces régions (Bhattacharyya, 1943).

L'estimation de la distribution d'un objet se fait toujours à partir de sa partie centrale. Selon les approches, on considérera simultanément les  $|P_i|$  parties indépendantes ou pas (voir la section 3). Les approches sont comparées essentiellement selon deux critères : le temps de traitement et l'erreur d'estimation. Cette erreur est donnée par la somme des distances euclidiennes entre chaque sommet des polygones et le sommet correspondant dans la vérité terrain. Nous comparons les performances de PS sans et avec permutation, qui seront respectivement dénotés PS et  $\overline{\overline{PS}}$ , de l'APF classique, avec 1 ou 2 couches, et de l'APF avec permutation avec 1 ou 2 couches dénotés respectivement  $APF_1$  et  $APF_2$ ,  $\overline{\overline{APF_1}}$  et  $\overline{\overline{APF_2}}$ . L'ensemble des résultats présentés dans cet article sont une moyenne sur 20 lancés.

## 4.2. Cas de l'estimation d'une seule densité

### 4.2.1. Erreurs d'estimation

Nous considérons d'abord le cas d'objets avec  $|P_i| = 4$  (voir un exemple figure 4.(a)), et nous nous intéressons à l'erreur d'estimation en fonction du nombre total  $P$  de parties à traiter. Le nombre de particules est choisi tel que  $N = 10 \times P$ . Le tableau 1 rend compte des résultats obtenus. Nous pouvons noter que, pour  $P$  fixé,  $\overline{\overline{PS}}$  donne des erreurs plus petites que PS, pour des temps de traitement plus faibles, tout comme  $\overline{\overline{APF_1}}$  et  $\overline{\overline{APF_2}}$  sont toujours meilleurs que  $APF_1$  et  $APF_2$ . En particulier, on note que  $\overline{\overline{APF_1}}$  fournit de meilleures estimations que  $APF_2$ . Ainsi, notre approche visant à ajouter une permutation, permet à la fois de diminuer l'erreur et les temps de calcul. Un dernier aspect important concerne le fait que l'écart de temps de calcul entre les versions avec et sans permutation augmente avec  $P$ . On peut ainsi remarquer que pour  $P = 17$ ,  $\overline{\overline{PS}}$  est 15 % plus rapide que PS, et  $\overline{\overline{APF_1}}$  est 4 % plus rapide que  $APF_1$ . Cette diminution du temps de traitement est due au fait qu'en traitant simultanément  $|P_i|$  parties, nous diminuons le nombre de rééchantillonnages nécessaires (puisque les traitements se font en parallèle). Cette étude sera l'objet de la section 4.2.3.

La méthode d'optimisation du recuit simulé intégrée dans le PF, donnant APF, permet de diminuer le nombre de particules. Nous proposons de comparer les erreurs d'estimation de la densité d'un objet avec  $|P_i| = 3$  et  $P = 10$  ( $|\mathcal{X}| = 30$ ), cf. la figure 5.b. Nous avons comparé PS et  $\overline{\overline{PS}}$  avec  $N = 120$ , et  $APF_1$ ,  $APF_2$ ,  $\overline{\overline{APF_1}}$  et

**Tableau 1.** Erreurs d'estimation (pixels) et temps de calcul (sec.), pour un objet avec  $|P_i| = 4$ , en faisant varier  $P$ 

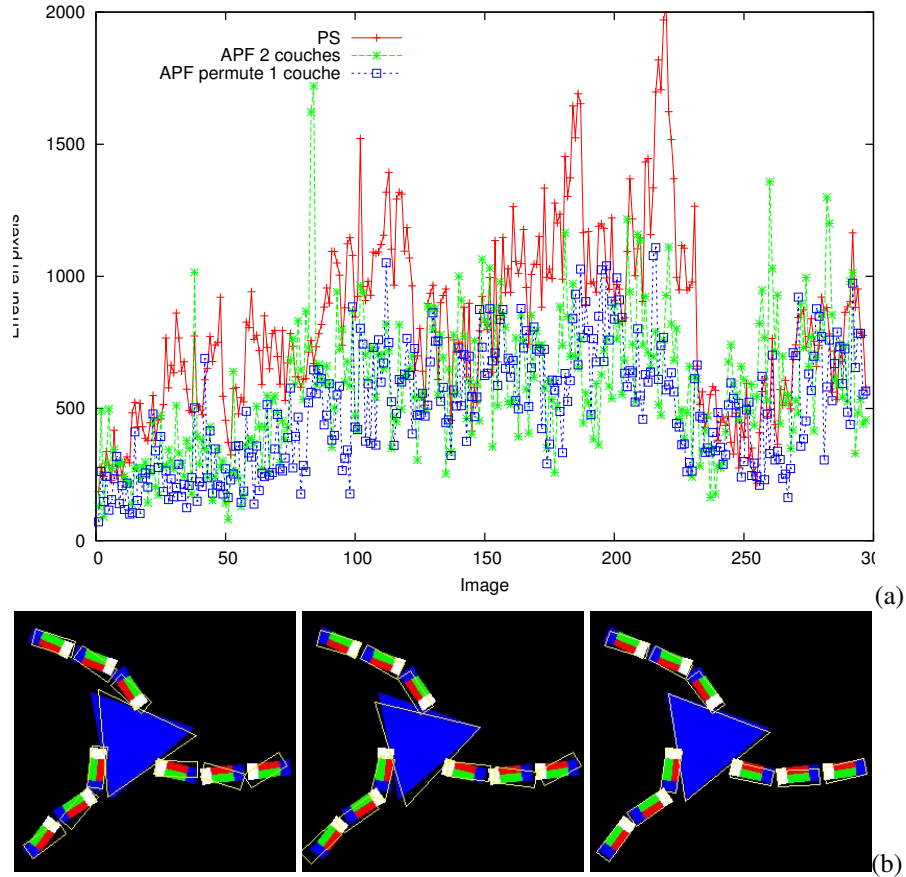
		PS	$\overline{\overline{PS}}$	APF <sub>1</sub>	APF <sub>2</sub>	$\overline{\overline{APF_1}}$	$\overline{\overline{APF_2}}$
P=5	erreur	226	196	190	170	160	154
	temps	3.5	3.5	6.4	9.3	6.2	9.3
P=9	erreur	278	207	234	217	193	177
	temps	11.2	10.2	20.9	31.1	19.9	30.4
P=13	erreur	693	567	549	516	499	478
	temps	23.9	21.5	48.1	66.5	44	62.3
P=17	erreur	685	651	630	546	534	511
	temps	46.4	39.1	87.6	125.3	84.2	120.9

$\overline{\overline{APF_2}}$  avec  $N = 40$ . Pour plus de lisibilité, nous avons reporté les courbes d'erreurs d'estimation de PS, APF<sub>2</sub> et  $\overline{\overline{APF_1}}$  sur la figure 5.(a). On remarque que  $\overline{\overline{APF_1}}$  est la plus stable des trois approches, alors qu'il n'utilise qu'une couche. Un résultat de suivi est donné sur la figure 5.(b) (le modèle estimé est dessiné en trait jaune sur l'image). Le tableau 2 reporte les erreurs moyennes de suivi et les temps de calcul pour toutes les méthodes. On remarque que  $\overline{\overline{APF_1}}$  produit de meilleures estimations que APF<sub>1</sub> et APF<sub>2</sub>, et ce en moins de temps de calcul que PS. Enfin, notre approche fait converger APF après une seule couche de recuit simulé, ce qui n'est pas le cas de la version sans permutation.

**Tableau 2.** Erreur d'estimation (pixels) et temps de traitement (sec.) de la densité de l'objet de la figure 5

	PS	$\overline{\overline{PS}}$	APF <sub>1</sub>	APF <sub>2</sub>	$\overline{\overline{APF_1}}$	$\overline{\overline{APF_2}}$
$N$	120	120	40	40	40	40
Erreur	818	627	799	581	496	493
Temps	23.9	23.9	21.2	30	20.9	30.5

Un des principaux avantages de notre approche est sa capacité à utiliser les indépendances dans les espaces probabilistes pour décomposer les calculs et ainsi les accélérer. Nous testons donc l'influence de  $|P_i|$  sur l'erreur d'estimation : les résultats sont reportés dans le tableau 3, pour  $N = 50$  et  $|\mathcal{X}| = 3(|P_i| + 1)$ . Avec ce nombre faible de particules, les estimations se dégradent considérablement avec l'augmentation de la taille de l'espace d'état traité (voir  $|P_i| = 8, 9, 10$ ). Toutefois, l'erreur d'estimation augmente moins fortement pour les méthodes avec permutation que pour celles sans permutation.



**Figure 5.** Estimation de la densité d'un objet avec  $|P_i| = 3$  et  $P = 10$  ( $|\mathcal{X}| = 30$ ). Comparaison entre PS avec  $N = 120$  particules,  $APF_2$  et  $\overline{APF}_1$  avec  $N = 40$ . Haut : courbes d'erreur d'estimation image par image. Bas : un résultat de suivi, de gauche à droite, PS,  $APF_2$ ,  $\overline{APF}_1$ . L'objet estimé est surimposé en trait jaune

#### 4.2.2. Tests de convergence

Dans la section 4.2.1, nous ne nous sommes pas posé la question du nombre optimal  $N$  de particules. Nous comparons donc maintenant les qualités d'estimation de la densité d'un objet avec  $P = 9$  et  $|P_i| = 4$  ( $|\mathcal{X}| = 27$ , voir figure 4.(a)) en fonction de  $N$ . Les courbes de convergence sont données dans la figure 6, selon la quantité de bruit gaussien présent dans les images de la séquence traitée ( $\sigma = \{0, 10, 20\}$ ). Nous pouvons voir que la permutation permet de diminuer significativement les erreurs d'estimation (l'échelle des ordonnées est logarithmique) ainsi que la rapidité de

**Tableau 3.** Erreurs (pixels) d'estimation de la densité en fonction de  $|P_i|$ , pour  $N = 50$  et  $|\mathcal{X}| = 3(|P_i| + 1)$ 

	PS	$\overline{\overline{PS}}$	APF <sub>1</sub>	APF <sub>2</sub>	$\overline{\overline{APF_1}}$	$\overline{\overline{APF_2}}$
$ P_i  = 2$	176	151	148	124	123	118
$ P_i  = 3$	180	170	152	140	126	119
$ P_i  = 4$	178	169	157	133	134	124
$ P_i  = 5$	341	293	262	271	238	214
$ P_i  = 6$	343	316	282	254	243	224
$ P_i  = 7$	467	417	432	391	379	342
$ P_i  = 8$	751	680	667	687	540	499
$ P_i  = 9$	488	417	422	376	353	332
$ P_i  = 10$	851	764	816	799	689	621

convergence. En particulier, on peut voir que  $\overline{\overline{APF_1}}$  et  $\overline{\overline{APF_2}}$  donnent des erreurs plus petites et convergent plus vite que l'APF classique. Nous pouvons également constater que  $\overline{\overline{PS}}$  est quasi-équivalent à APF<sub>1</sub> : cela montre que notre approche est autant, sinon plus, robuste qu'une méthode d'optimisation telle que celle présente dans APF (le recuit simulé). Cela s'accroît avec l'augmentation du bruit dans la séquence :  $\overline{\overline{PS}}$  devient meilleur que APF et converge tout aussi rapidement.

Enfin, le tableau 4 donne les erreurs d'estimation et les temps de calcul en fonction du nombre de couches utilisées dans APF et  $\overline{\overline{APF}}$  pour  $N = 20$  et  $|\mathcal{X}| = 30$ . Là encore, notre approche reste plus robuste.

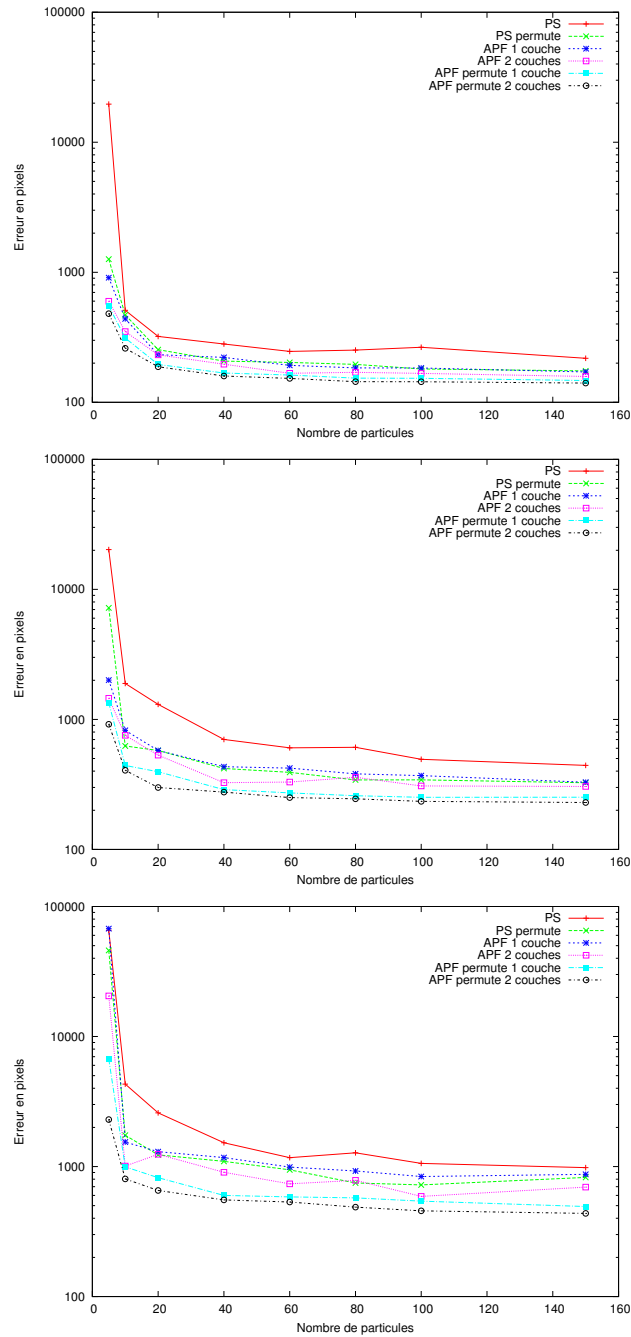
**Tableau 4.** Erreur d'estimation de la densité (pixels) et temps de calcul (sec.) pour  $P = 10$ ,  $|P_i| = 4$  ( $|\mathcal{X}| = 30$ ). Variation du nombre  $L$  de couches pour APF et  $\overline{\overline{APF}}$ , avec  $N = 20$ 

$L$		1	2	3	4	5	6	7
APF	erreur	707	594	516	532	551	542	544
	temps	5.6	8.6	11.2	14.2	17.8	19.9	22.1
$\overline{\overline{APF}}$	erreur	491	461	473	448	420	419	420
	temps	5.6	8.6	11.4	14.3	17.7	20	22.4

#### 4.2.3. Temps de calcul

Il y a principalement quatre opérations qui interviennent dans le temps de calcul global : le temps de propagation des particules et de calcul de la vraisemblance pour l'étape de correction, qui sont similaires pour toutes les approches, et les temps de rééchantillonnage (incluant la normalisation des poids) et de permutation qui sont spécifiques à chaque approche. Pour que notre méthode soit intéressante, il faut que l'introduction de la permutation ne ralentisse pas les temps de calcul, donc que le





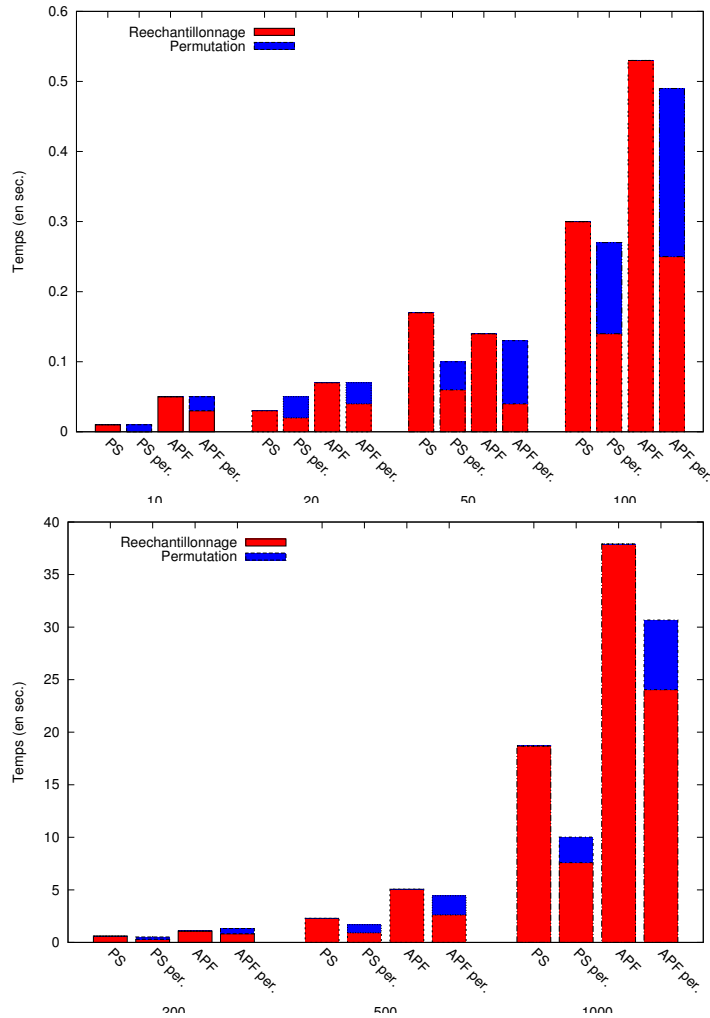
**Figure 6.** Tests de convergence. Erreur d'estimation de la densité d'un objet pour  $P = 9$ ,  $|P_i| = 4$  (échelle logarithmique) en fonction de  $N$  pour PS,  $\overline{PS}$ ,  $APF_1$ ,  $APF_2$ ,  $\overline{APF_1}$  et  $\overline{APF_2}$ , selon la quantité de bruit gaussien présent dans la séquence vidéo, de haut en bas :  $\sigma = 0$ ,  $\sigma = 10$  et  $\sigma = 20$

temps de rééchantillonnage plus celui de permutation soit au plus équivalent au temps de rééchantillonnage des approches classiques (PS et APF). Nous avons mesuré ces temps en prenant en compte différents critères. La figure 7 montre les temps, en fonction de  $N$ , pour l'estimation séquentielle de la densité d'un objet avec  $K = 6$  et  $P = 6$  ( $|\mathcal{X}| = 18$ ). On remarque que notre approche réduit bien les temps de rééchantillonnage (en rouge), et que l'ajout de la permutation (en bleu) n'entraîne pas un surcoût. De plus, avec l'augmentation de  $N$ , notre approche diminue les temps de calcul. Par exemple, pour  $N = 1\,000$ , les temps totaux de traitement pour PS,  $\overline{\overline{PS}}$ ,  $\overline{\overline{APF_1}}$  et  $\overline{\overline{APF_1}}$  sont respectivement de 163, 157, 349 et 329 secondes, soit un gain de 4 à 6 %. La figure 8.(a) montre les résultats en faisant varier  $|P_i|$ , avec  $N = 200$  et  $|\mathcal{X}| = 3(|P_i| + 1)$ . On remarque que le temps de rééchantillonnage pour  $\overline{\overline{PS}}$  et  $\overline{\overline{APF_1}}$  diminue avec  $|P_i|$ , de même que le temps total de traitement. Par exemple, pour  $|P_i| = 10$  ( $|\mathcal{X}| = 33$ ), les temps totaux de traitement pour PS,  $\overline{\overline{PS}}$ ,  $\overline{\overline{APF_1}}$  et  $\overline{\overline{APF_1}}$  sont respectivement de 51, 50, 109 et 105 secondes, soit un gain de 2 à 4 %. Enfin, la figure 8.(b) montre les résultats en faisant varier  $P$ , avec  $|P_i| = 5$  et  $N = 200$  (i.e. on fait varier  $K$ , et  $|\mathcal{X}| = 3((K - 1)|P_i| + 1)$ ). Là encore, on constate que les temps diminuent avec l'augmentation de la longueur du bras. Pour  $K = 7$  ( $|\mathcal{X}| = 93$ ), les temps totaux de traitement pour PS,  $\overline{\overline{PS}}$ ,  $\overline{\overline{APF_1}}$  et  $\overline{\overline{APF_1}}$  sont respectivement de 106, 94, 224 et 207 secondes, soit un gain de 7 à 11 %.

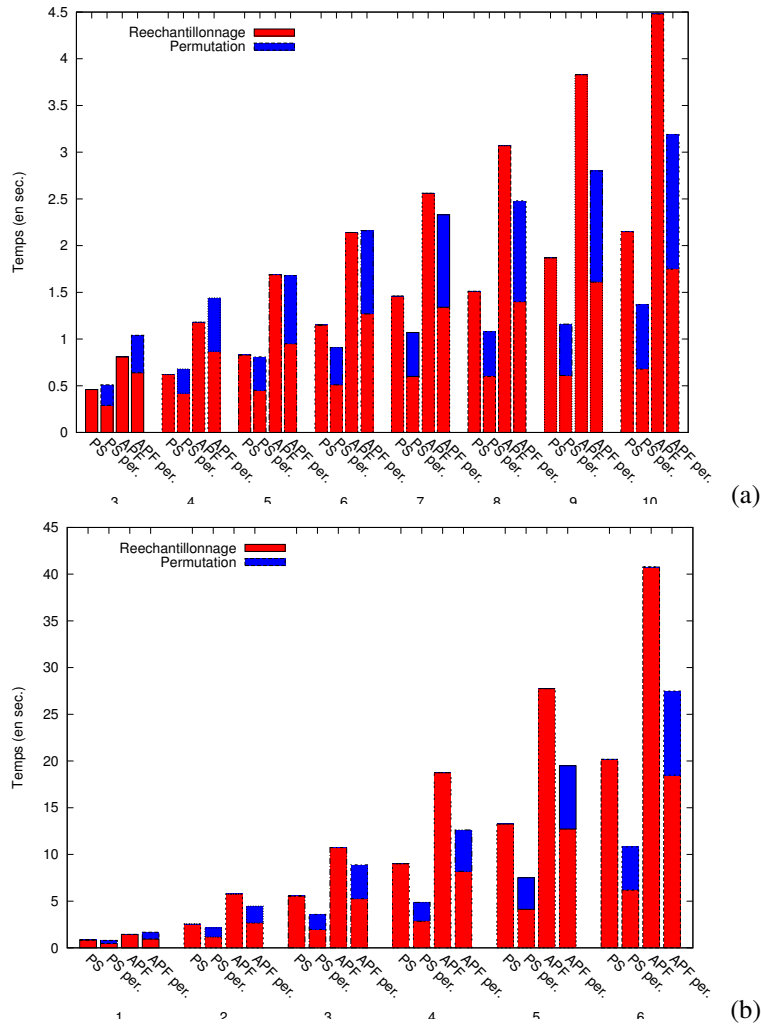
### 4.3. Extension à l'estimation multiple

Dans cette section, nous traitons le cas où l'on a plusieurs distributions à estimer (par exemple, en vision cela correspond à un cas de suivi multiobjets). Généralement, il existe deux manières de procéder : on utilise un filtre par densité à estimer ou un filtre pour l'ensemble. Dans ce dernier cas, on se retrouve à avoir à gérer de très grands espaces d'état puisque, pour estimer  $M$  densités, la taille des vecteurs d'état est multipliée par  $M$ . Le but de cette section est de montrer que notre méthode est adaptée à la gestion des grands espaces d'état en prenant en compte les indépendances probabilistes, et qu'elle est ainsi capable de les décomposer naturellement.

Le tableau 5 donne les résultats comparatifs d'erreurs d'estimation et de temps de traitement pour PS et  $\overline{\overline{PS}}$ . Ici, chaque objet suivi est défini par  $|P_j| = 4$  et  $K = 3$ , et l'ensemble des objets bougent et se déforment indépendamment. Nous avons testé le cas avec  $M = 2$  et  $M = 3$  objets. Quand un seul filtre est utilisé pour l'estimation des densités de l'ensemble des objets, nous utilisons systématiquement  $N = M \times 100$  particules. Logiquement, dans l'autre cas, où un filtre est dédié à l'estimation d'un seul objet, le nombre de particules nécessaires devrait être plus petit. Cependant, nous testons l'estimation soit avec  $N = 100$ , soit avec  $N = M \times 100$  particules par filtre. Nous remarquons tout d'abord que les erreurs d'estimation les plus basses sont obtenues avec notre  $\overline{\overline{PS}}$ , et ce que cela soit avec un seul filtre pour l'ensemble des densités à estimer, ou pour un filtre par densité. On remarque également que lorsqu'on utilise  $M$  filtres, le filtre PS est meilleur qu'avec un unique filtre. C'est le cas quel



**Figure 7.** Temps de rééchantillonnage (en rouge) et de permutation (en bleu) selon les approches  $(PS, \overline{PS}, APF_1, \overline{APF}_1)$ , en faisant varier  $N$  ( $K = 6, P = 6, |\mathcal{X}| = 18$ )



**Figure 8.** Temps de rééchantillonnage (en rouge) et de permutation (en bleu) selon les approches ( $PS$ ,  $\overline{PS}$ ,  $APF_1$ ,  $\overline{APF}_1$ ), en faisant varier (a)  $|P_i|$  ( $N = 200$ ,  $|\mathcal{X}| = 3(|P_i| + 1)$ ) et (b) la longueur ( $K - 1$ ) des bras ( $|P_i| = 5$ ,  $N = 200$ ,  $|\mathcal{X}| = 3((K - 1)|P_i| + 1)$ ).

que soit le nombre  $N$  de particules utilisées. Cela montre que PS est meilleur pour estimer des densités définies dans des espaces de plus petites dimension. Avec  $\overline{\overline{PS}}$ , au contraire, les erreurs d'estimation sont plus petites quand on utilise un filtre plutôt que  $M$ . Cela est dû au fait que l'augmentation de la taille de l'espace d'état augmente également l'efficacité de la procédure de permutation, puisque le produit des meilleurs  $w^{(i),k}$  a aussi tendance à augmenter.

Concernant les temps de réponse, pour un nombre fixé de particules, les temps de rééchantillonnage de PS sont divisés par  $M$  lorsqu'on utilise un seul filtre pour estimer les  $M$  densités. Cela vient du fait que, bien que le nombre de rééchantillonnages soit identique pour les deux méthodes, la dimension de l'espace d'état traité pour un seul filtre est  $M$  fois plus grande que celle dans le cadre de filtres multiples. En revanche, les temps de rééchantillonnage sont les mêmes avec  $\overline{\overline{PS}}$  et ce, que l'on utilise un seul ou  $M$  filtres. En fait, les  $M$  filtres font bien  $M$  fois plus de rééchantillonnages qu'un filtre unique, mais ce dernier est réalisé dans des espaces de dimension  $M$  fois plus grands, ce qui est équivalent en temps de calcul. Pour les mêmes raisons, les temps de permutation sont aussi équivalents. Enfin, nous remarquons que tous les temps de traitement diminuent avec  $N$ , alors que l'erreur d'estimation augmente, ce qui est logique et met bien en avant le compromis classique entre les temps de réponse et la qualité de l'estimation.

Globalement, notre approche donne des résultats plus précis que le filtre PS tout en étant plus rapide. De plus, d'un point de vue efficacité, il est notable qu'un seul filtre  $\overline{\overline{PS}}$ , travaillant par conséquent dans un espace de plus grande dimension, donne des erreurs d'estimation plus petites que lorsqu'on utilise  $M$  filtres, soit un par objet. Cela montre à quel point notre approche est adaptée pour gérer les grands espaces d'état, en décomposant de manière efficace les calculs.

**Tableau 5.** Erreurs d'estimation ( $e$ , en pixels) et temps de traitement (en secondes,  $t$  temps total de traitement,  $r$  temps de rééchantillonnage et  $p$  temps de permutation), pour l'estimation des densités de  $M$  objets avec  $K = 3$  et  $|P_i| = 4$  ( $N$  est le nombre de particules utilisées par chaque filtre)

	$N =$	1 PS	$M$ PS		1 $\overline{\overline{PS}}$	$M$ $\overline{\overline{PS}}$	
		$M \times 100$	$M \times 100$	100	$M \times 100$	$M \times 100$	100
$M = 2$	$e$	356	281	311	211	236	260
	$t$	36.6	34.9	16.4	36.3	34.9	16.3
	$r$	3.0	1.7	0.6	0.9	1.0	0.3
	$p$	-	-	-	1.1	0.5	0.2
$M = 3$	$e$	611	312	370	242	253	308
	$t$	86.9	78.6	37.4	82.1	79.8	37.8
	$r$	12.3	4.5	1.6	2.2	2.5	0.8
	$p$	-	-	-	3.0	1.3	0.5

## 5. Conclusion

Nous avons présenté dans cet article une nouvelle approche d'estimation séquentielle de densité qui vise (i) à exploiter les indépendances probabilistes encodées par les DBN afin de ne réaliser les calculs du filtre particulaire que dans des sous-espaces de petite dimension et (ii) faire des permutations de certains sous-ensembles de particules de manière à se concentrer sur les modes de la distribution à estimer. Nous avons proposé un cadre théorique rigoureux qui nous garantit que la distribution ainsi estimée n'est pas modifiée, ainsi qu'une analyse en complexité au pire cas. Les tests ont permis de montrer que notre approche, intégrant une étape supplémentaire par rapport aux autres (la permutation), n'est pas plus coûteuse. En effet, le nombre d'étapes de rééchantillonnage nécessaires est diminué avec le traitement simultané des parties indépendantes de la densité de l'objet à estimer. Ce gain de temps permet d'absorber le surcoût dû à l'étape de permutation, et ainsi de ne pas augmenter les temps globaux de traitement. De plus, nous avons montré que plus la taille de l'espace traité augmente, plus notre approche permet de gagner du temps. Nos travaux actuels portent sur la définition d'un critère plus efficace pour la permutation des sous-ensembles de particules, en particulier à définir ce qu'est (ou pas) un bon nuage de particules. Enfin, l'introduction des modèles de Markov couples et triplet (Derrode *et al.*, 2004) permettrait de s'affranchir des fortes hypothèses classiques et ainsi améliorer les résultats obtenus.

## 6. Appendice : démonstrations

Pour démontrer mathématiquement que les particules produites par  $\overline{PS}$  ou  $\overline{APF}$  constituent un échantillon représentatif de la densité *a posteriori*  $p$ , il convient de définir précisément les différentes densités estimées à chaque étape de ces algorithmes. Pour cela, étant donné que l'on ne propage pas toutes les parties des particules en même temps, il convient de différencier les celles que l'on propage dans l'algorithme 5 à la  $j$ ème étape,  $P_j$ , l'ensemble de celles que l'on a déjà propagées,  $Q_j = \sum_{h=1}^j P_h$ , et celles qu'il reste à propager,  $R_j = \sum_{h=j+1}^K P_h$ . Par abus de notation, nous poserons  $Q_0 = \emptyset$  et  $R_K = \emptyset$ .

**Lemme 1** *L'ensemble de particules produites par l'algorithme 5 sans l'étape de permutation représente  $p(\mathbf{x}_{t+1}|\mathbf{y}_{1:t+1})$ .*

**Démonstration:** Par récurrence sur  $j$ . Supposons qu'avant de traiter les parties de  $P_j$ , l'ensemble de particules courant estime la densité  $p(\mathbf{x}_{t+1}^{Q_{j-1}}, \mathbf{x}_t^{R_{j-1}}|\mathbf{y}_{1:t}, \mathbf{y}_{t+1}^{Q_{j-1}})$ . Cette propriété est clairement vérifiée pour  $j = 1$  puisque, avant de traiter l'ensemble de parties  $P_1$ , l'ensemble de particules estime  $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ . Examinons maintenant successivement la densité estimée par l'ensemble de particules obtenu après l'application de l'étape de propagation en parallèle sur les parties de  $P_j$ , puis celle après la phase de correction et, enfin, celle après rééchantillonnage.

1. Montrons qu'après la propagation en parallèle des parties de  $P_j$  (étape de prédiction), l'ensemble de particules représente la densité  $p(\mathbf{x}_{t+1}^{Q_j}, \mathbf{x}_t^{R_j} | \mathbf{y}_{1:t}, \mathbf{y}_{t+1}^{Q_{j-1}})$ . Toutes ces opérations en parallèle correspondent au calcul de la quantité :

$$\int p(\mathbf{x}_{t+1}^{Q_{j-1}}, \mathbf{x}_t^{R_{j-1}} | \mathbf{y}_{1:t}, \mathbf{y}_{t+1}^{Q_{j-1}}) \prod_{k \in P_j} p(\mathbf{x}_{t+1}^k | \mathbf{Pa}(\mathbf{x}_{t+1}^k)) d\mathbf{x}_t^{P_j}.$$

Par  $d$ -séparation, un nœud est indépendant de tous ses non descendants conditionnellement à ses parents. Par conséquent, pour tout  $k \in P_j$ ,  $\mathbf{x}_{t+1}^k$  est indépendant de  $\mathbf{x}_{t+1}^{P_j \setminus \{k\}} \cup \mathbf{x}_{t+1}^{Q_{j-1}} \cup \mathbf{x}_t^{R_{j-1}} \cup \mathbf{y}_{1:t} \cup \mathbf{y}_{t+1}^{Q_{j-1}}$  conditionnellement à  $\mathbf{Pa}(\mathbf{x}_{t+1}^k)$ . Donc l'intégrale ci-dessus est équivalente à :

$$\begin{aligned} & \int p(\mathbf{x}_{t+1}^{Q_{j-1}}, \mathbf{x}_t^{R_{j-1}} | \mathbf{y}_{1:t}, \mathbf{y}_{t+1}^{Q_{j-1}}) \\ & p(\mathbf{x}_{t+1}^{P_j} | \mathbf{x}_{t+1}^{Q_{j-1}}, \mathbf{x}_t^{R_{j-1}}, \mathbf{y}_{1:t}, \mathbf{y}_{t+1}^{Q_{j-1}}) d\mathbf{x}_t^{P_j} \\ & = \int p(\mathbf{x}_{t+1}^{P_j}, \mathbf{x}_{t+1}^{Q_{j-1}}, \mathbf{x}_t^{R_{j-1}} | \mathbf{y}_{1:t}, \mathbf{y}_{t+1}^{Q_{j-1}}) d\mathbf{x}_t^{P_j}. \end{aligned}$$

Etant donné que  $Q_j = Q_{j-1} \cup P_j$  et  $R_{j-1} = P_j \cup R_j$ , l'équation ci-dessus est équivalente à  $p(\mathbf{x}_{t+1}^{Q_j}, \mathbf{x}_t^{R_j} | \mathbf{y}_{1:t}, \mathbf{y}_{t+1}^{Q_{j-1}})$ .

2. Montrons maintenant qu'après les corrections menées en parallèle des parties appartenant à  $P_j$ , l'ensemble de particules estime la densité  $p(\mathbf{x}_{t+1}^{Q_j}, \mathbf{x}_t^{R_j} | \mathbf{y}_{1:t}, \mathbf{y}_{t+1}^{Q_j})$ . Ces opérations correspondent au calcul, à une constante proportionnelle près, de la densité :

$$p(\mathbf{x}_{t+1}^{Q_j}, \mathbf{x}_t^{R_j} | \mathbf{y}_{1:t}, \mathbf{y}_{t+1}^{Q_{j-1}}) \times \prod_{k \in P_j} p(\mathbf{y}_{t+1}^k | \mathbf{x}_{t+1}^k).$$

Par  $d$ -séparation, les nœuds  $\mathbf{y}_{t+1}^k$  sont indépendants du reste du DBN conditionnellement à  $\mathbf{x}_{t+1}^k$ . Donc  $p(\mathbf{y}_{t+1}^{P_j} | \mathbf{x}_{t+1}^{Q_j}, \mathbf{x}_t^{R_j}, \mathbf{y}_{1:t}, \mathbf{y}_{t+1}^{Q_{j-1}}) = \prod_{k \in P_j} p(\mathbf{y}_{t+1}^k | \mathbf{x}_{t+1}^k)$ . Après les corrections sur les parties de  $P_j$ , l'ensemble de particules estime donc la densité  $p(\mathbf{x}_{t+1}^{Q_j}, \mathbf{x}_t^{R_j}, \mathbf{y}_{t+1}^{P_j} | \mathbf{y}_{1:t}, \mathbf{y}_{t+1}^{Q_{j-1}})$ , qui, après normalisation, est égale à  $p(\mathbf{x}_{t+1}^{Q_j}, \mathbf{x}_t^{R_j} | \mathbf{y}_{1:t}, \mathbf{y}_{t+1}^{Q_{j-1}}, \mathbf{y}_{t+1}^{P_j}) = p(\mathbf{x}_{t+1}^{Q_j}, \mathbf{x}_t^{R_j} | \mathbf{y}_{1:t}, \mathbf{y}_{t+1}^{Q_j})$ . Etant donné que les ré-échantillonnages n'affectent pas les densités, à la fin de l'algorithme, l'ensemble de particules estime la densité  $p(\mathbf{x}_{t+1}^{Q_K}, \mathbf{x}_t^{R_K} | \mathbf{y}_{1:t}, \mathbf{y}_{t+1}^{Q_K}) = p(\mathbf{x}_{t+1} | \mathbf{y}_{1:t+1})$ .  $\square$

**Démonstration de la proposition 1:** Soit  $k \in P_j$ , appelons  $\text{Desc}_t(\mathbf{x}_t^k)$  l'ensemble des descendants de  $\mathbf{x}_t^k$  dans la tranche de temps  $t$ . Considérons une permutation  $\sigma$  admissible pour l'opération  $\rightleftharpoons^{P_j}$ . Montrons que l'ensemble de particules estime la même densité avant et après l'application de cette permutation.

Si  $j = 1$ , c'est-à-dire si l'on applique, dans l'algorithme 5, l'opération  $\rightleftharpoons^{P_1}$  sur  $P_1$  (et ses descendants), la proposition est trivialement vérifiée puisque, dans ce cas,  $\sigma$  est appliquée sur tous les nœuds du DBN, ce qui revient simplement à faire une permutation des particules dans l'ensemble de particules.

Supposons maintenant que  $j \neq 1$  et considérons l'application de  $\sigma$  sur les nœuds  $\{\mathbf{x}_{t+1}^k\} \cup \text{Desc}_t(\mathbf{x}_t^k)$ . Lorsque l'on traite les permutations des parties de  $P_j$ , les particules sont du type  $(\mathbf{x}_{t+1}^{Q_j}, \mathbf{x}_t^{R_j})$ . Partitionnons ces parties de la manière suivante :

$$\begin{aligned} - \mathbf{x}_t^D &= \text{Desc}_t(\mathbf{x}_t^k), \\ - \mathbf{x}_{t+1}^{k'} &= \text{pa}(\mathbf{x}_{t+1}^k), \\ - \mathbf{x}_{t+1}^V &= \mathbf{x}_{t+1}^{Q_j} \setminus (\{\mathbf{x}_{t+1}^k, \mathbf{x}_{t+1}^{k'}\}) \text{ et} \\ - \mathbf{x}_t^W &= \mathbf{x}_t^{R_j} \setminus \mathbf{x}_t^D, \end{aligned}$$

c'est-à-dire que  $\mathbf{x}_{t+1}^k \cup \mathbf{x}_t^D$  sont les parties permutées,  $\mathbf{x}_{t+1}^{k'}$  représente la partie qui est identique pour les particules permutées, et  $\mathbf{x}_{t+1}^V \cup \mathbf{x}_t^W$  représentent les parties qui ne sont pas concernées par les permutations. Alors :

$$\begin{aligned} p(\mathbf{x}_{t+1}^{Q_j}, \mathbf{x}_t^{R_j} | \mathbf{y}_{1:t}, \mathbf{y}_{t+1}^{Q_j}) &\propto p(\mathbf{x}_{t+1}^{Q_j}, \mathbf{x}_t^{R_j}, \mathbf{y}_{1:t}, \mathbf{y}_{t+1}^{Q_j}) \\ &= p(\mathbf{x}_{t+1}^{\{k,k'\} \cup V}, \mathbf{x}_t^{D \cup W}, \mathbf{y}_{1:t+1}^{\{k,k'\} \cup V}, \mathbf{y}_{1:t}^{D \cup W}) \\ &= \int p(\mathbf{x}_{t+1}^{\{k\} \cup V}, \mathbf{x}_{1:t+1}^{k'}, \mathbf{x}_t^{D \cup W}, \mathbf{y}_{1:t+1}^{\{k,k'\} \cup V}, \mathbf{y}_{1:t}^{D \cup W}) d\mathbf{x}_{1:t}^{k'} \end{aligned}$$

Conditionnellement à  $\{\mathbf{x}_{1:t+1}^{k'}\}$ ,  $S = \{\mathbf{x}_{t+1}^k\} \cup \mathbf{x}_t^D \cup \mathbf{y}_{1:t+1}^k \cup \mathbf{y}_{1:t}^D$  est indépendant du reste du DBN car, d'après la définition 2, aucune chaîne active ne peut passer par un arc sortant d'un nœud appartenant à l'ensemble conditionnant et, en supprimant du DBN les arcs sortants des nœuds  $\{\mathbf{x}_{1:t+1}^{k'}\}$ ,  $S$  n'est plus connecté au reste du DBN. Pour la même raison,  $\mathbf{x}_{t+1}^V \cup \mathbf{x}_t^W \cup \mathbf{y}_{1:t+1}^V \cup \mathbf{y}_{1:t}^W$  est indépendant du reste du DBN conditionnellement à  $\{\mathbf{x}_{1:t+1}^{k'}\}$ . Par conséquent, l'intégrale ci-dessus est égale à :

$$\begin{aligned} \int p(\mathbf{x}_{1:t+1}^{k'}, \mathbf{y}_{1:t+1}^{k'}) p(\mathbf{x}_{t+1}^k, \mathbf{x}_t^D, \mathbf{y}_{1:t+1}^k, \mathbf{y}_{1:t+1}^D | \mathbf{x}_{1:t+1}^{k'}) \\ p(\mathbf{x}_{t+1}^V, \mathbf{x}_t^W, \mathbf{y}_{1:t+1}^V, \mathbf{y}_{1:t}^W | \mathbf{x}_{1:t+1}^{k'}) d\mathbf{x}_{1:t}^{k'} \end{aligned} \quad [5]$$

Permuter les particules sur les parties  $\{\mathbf{x}_{t+1}^k\} \cup \mathbf{x}_t^D$  pour des valeurs fixées de  $\mathbf{x}_{1:t+1}^{k'}$  ne peut pas changer l'estimation de la densité  $p(\mathbf{x}_{t+1}^k, \mathbf{x}_t^D, \mathbf{y}_{1:t+1}^k, \mathbf{y}_{1:t}^D | \mathbf{x}_{1:t+1}^{k'})$  car les estimations par échantillon sont insensibles à l'ordre des éléments dans l'échantillon. De plus, une telle permutation ne peut pas non plus modifier l'estimation de la densité  $p(\mathbf{x}_{t+1}^V, \mathbf{x}_t^W, \mathbf{y}_{1:t+1}^V, \mathbf{y}_{1:t}^W | \mathbf{x}_{1:t+1}^{k'})$  puisque  $\mathbf{x}_{t+1}^V \cup \mathbf{x}_t^W \cup \mathbf{y}_{1:t+1}^V \cup \mathbf{y}_{1:t}^W$  est indépendant de  $\{\mathbf{x}_{t+1}^k\} \cup \mathbf{x}_t^D$  conditionnellement à  $\{\mathbf{x}_{1:t+1}^{k'}\}$ . Par conséquent, appliquer la permutation  $\sigma$  sur les parties de l'objet  $\{\mathbf{x}_{t+1}^k\} \cup \mathbf{x}_t^D$  ne peut affecter l'estimation de l'équation (5) et, *a fortiori*, l'estimation de la densité  $p(\mathbf{x}_{t+1}^{Q_j}, \mathbf{x}_t^{R_j} | \mathbf{y}_{1:t}, \mathbf{y}_{t+1}^{Q_j})$ .  $\square$

## 7. Bibliographie

Ahmadi M., Mojallali H., Izadi-Zamanabadi R., « State estimation of nonlinear stochastic systems using a novel meta-heuristic particle filter », *Swarm and Evolutionary Computation*, vol. 4, p. 44-53, 2012.



- Besada-Portas E., Plis S., Cruz J., Lane T., « Parallel Subspace Sampling for Particle Filtering in Dynamic Bayesian Networks », *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD)*, p. 131-146, 2009.
- Bhattacharyya A., « On a measure of divergence between two statistical populations defined by probability distributions », *Bulletin of the Calcutta Mathematical Society*, vol. 35, p. 99-110, 1943.
- Bray M., Koller-Meier E., Müller P., Schraudolph N. N., Gool L. V., « Stochastic Optimization for High-Dimensional Tracking in Dense Range Maps », *IEE Proceedings Vision, Image and Signal Processing*, vol. 152, n° 4, p. 501-512, 2005.
- Comaniciu D., Ramesh V., Meer P., « Kernel-based object tracking », *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 25, p. 564-577, 2003.
- Derrode S., Pieczynski W., « Signal and image segmentation using Pairwise Markov chains », *IEEE Transactions on Signal Processing*, vol. 52, p. 2477-2489, 2004.
- Deutscher J., Reid I., « Articulated Body Motion Capture by Stochastic Search », *International Journal of Computer Vision*, vol. 61, p. 185-205, 2005.
- Douc R., Cappé O., Moulines E., « Comparison of resampling schemes for particle filtering », *Proceedings of the International Symposium on Image and Signal Processing and Analysis (ISPA)*, p. 64-69, 2005.
- Doucet A., de Freitas N., Murphy K., Russell S., « Rao-Blackwellised Particle Filtering for Dynamic Bayesian Networks », *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI)*, p. 176-183, 2000.
- Gonzales C., Dubuisson S., Nguyen X.-S., « Simultaneous Partitioned Sampling for articulated object tracking », *Proceedings of Advanced Concepts for Intelligent Vision Systems (ACIVS)*, Lecture Notes in Computer Science, p. 150-161, 2011.
- Gordon N., Salmond D., Smith A., « Novel approach to nonlinear/non-Gaussian Bayesian state estimation », *IEE Proceedings of Radar and Signal Processing*, vol. 140, n° 2, p. 107-113, 1993.
- Han H., Ding Y.-S., Hao K.-R., Liang X., « An evolutionary particle filter with the immune genetic algorithm for intelligent video target tracking », *Computers and mathematics with application*, vol. 62, p. 185-205, 2005.
- John V., Trucco E., Ivekovic S., « Markerless human articulated tracking using hierarchical particle swarm optimisation », *Image and Vision Computing*, vol. 28, p. 1530-1547, 2010.
- Kanazawa K., Koller D., Russell S., « Stochastic Simulation Algorithms for Dynamic Probabilistic Networks », *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI)*, p. 346-35, 1995.
- Lucas B., Kanade T., « An iterative image registration technique with an application to stereo vision », *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, p. 674-679, 1981.
- MacCormick J., Blake A., « A probabilistic exclusion principle for tracking multiple objects », *Proceedings of the International Conference on Computer Vision (ICCV)*, p. 572-587, 1999.
- MacCormick J., Isard M., « Partitioned sampling, articulated objects, and interface-quality hand tracking », *Proceedings of the European Conference on Computer Vision (ECCV)*, p. 3-19, 2000.

- Murphy K., *Dynamic Bayesian Networks : Representation, Inference and Learning*, PhD thesis, UC Berkeley, Computer Science Division, 2002.
- Pantrigo J. J., Sánchez A., Montemayor A. S., Duarte A., « Multi-dimensional visual tracking using scatter search particle filter », *Pattern Recognition*, vol. 29, p. 1160-1174, 2008.
- Pearl J., *Probabilistic Reasoning in Intelligent Systems : Networks of Plausible Inference*, Morgan Kaufman Publishers, 1988.
- Rose C., Saboune J., Charpillet F., « Reducing particle filtering complexity for 3D motion capture using dynamic Bayesian networks », *Proceedings of the Conference on Artificial Intelligence (AAAI)*, p. 1396-1401, 2008.
- Sánchez A., Pantrigo J., Gianikellis K., « Combining Particle Filter and Population-based Metaheuristics for Visual Articulated Motion Tracking », *Electronic Letters on Computer Vision and Image Analysis*, vol. 5, n° 3, p. 68-83, 2005.
- Sudderth E., Ihler A., Isard M., Freeman W., Willsky A., « Nonparametric belief propagation », *Communications of ACM*, vol. 53, p. 95-103, 2010.
- Zhao J., Li Z., « Particle filter based on Particle Swarm Optimization resampling for vision tracking », *Expert Systems With Applications*, vol. 37, p. 8910-8914, 2010.

**ANNEXE POUR LE SERVICE FABRICATION**  
A FOURNIR PAR LES AUTEURS AVEC UN EXEMPLAIRE PAPIER  
DE LEUR ARTICLE ET LE COPYRIGHT SIGNE PAR COURRIER  
LE FICHER PDF CORRESPONDANT SERA ENVOYE PAR E-MAIL

1. ARTICLE POUR LA REVUE :

*Revue d'intelligence artificielle - n°1/2013*

2. AUTEURS :

*Séverine Dubuisson\* — Christophe Gonzales\**

3. TITRE DE L'ARTICLE :

*Estimation de densités non paramétriques  
et multimodales par permutation  
de sous-particules*

4. TITRE ABRÉGÉ POUR LE HAUT DE PAGE MOINS DE 40 SIGNES :

*Estimation de densités avec permutations*

5. DATE DE CETTE VERSION :

*12 février 2013*

6. COORDONNÉES DES AUTEURS :

– adresse postale :

\* Laboratoire LIP6 (CNRS - UPMC) — 4 place Jussieu, 75005 Paris  
severine.dubuisson@lip6.fr, christophe.gonzales@lip6.fr

– téléphone : 01 44 27 71 29

– télécopie : 01 44 27 53 53

– e-mail : Severine.Dubuisson@lip6.fr

7. LOGICIEL UTILISÉ POUR LA PRÉPARATION DE CET ARTICLE :

L<sup>A</sup>T<sub>E</sub>X, avec le fichier de style `article-hermes.cls`,  
version 1.23 du 17/11/2005.

8. FORMULAIRE DE COPYRIGHT :

Retourner le formulaire de copyright signé par les auteurs, téléchargé sur :  
<http://www.revuesonline.com>