



Poursuite multi-cibles sur signal brut

Audrey Cuillery

► To cite this version:

Audrey Cuillery. Poursuite multi-cibles sur signal brut. Traitement du signal et de l'image [eess.SP]. Université de Rennes, 2022. Français. NNT : 2022REN1S027 . tel-03850485v2

HAL Id: tel-03850485

<https://theses.hal.science/tel-03850485v2>

Submitted on 18 Nov 2022

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

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

THESE DE DOCTORAT DE

L'UNIVERSITE DE RENNES 1

ECOLE DOCTORALE N° 601

*Mathématiques et Sciences et Technologies
de l'Information et de la Communication*

Spécialité : *Signal, Image, Vision*

Par

Audrey CUILLERY

Poursuite multi-cibles sur signal brut

Thèse présentée et soutenue à Rennes, le 17 juin 2022

Unité de recherche : centre de recherche INRIA Rennes Bretagne Atlantique

Thèse N° :

Rapporteurs avant soutenance :

François Desbouvries professeur à Télécom Sud-Paris, Évry
Christian Musso maître de recherche à l'ONERA, Palaiseau

Composition du Jury :

Président :	François Septier	professeur à l'université de Bretagne Sud, Vannes
Examineurs :	Daniel Clark	maître de conférences à Télécom Sud-Paris, Évry
	François Desbouvries	professeur à Télécom Sud-Paris, Évry
	Salima El Kolei	enseignante-chercheuse à l'ENSAI, Bruz
	Christian Musso	maître de recherche à l'ONERA, Palaiseau
	François Septier	professeur à l'université de Bretagne Sud, Vannes
Directeur de thèse :	François Le Gland	directeur de recherche à l'INRIA, Rennes

Poursuite multi-cibles sur signal brut

Audrey Cuillery

Table des Matières

Introduction	3
1 Filtrage bayésien et solutions d'implémentation	9
1.1 Modélisation et filtrage bayésien optimal	9
1.2 Principe de Monte Carlo	14
1.3 Echantillonnage d'importance	16
1.4 Filtrage particulaire	19
1.4.1 Filtre particulaire et échantillonnage d'importance séquentiel	19
1.4.2 Filtre particulaire auxiliaire	24
1.4.3 Filtre particulaire marginalisé (ou <i>Rao-Blackwellisé</i>)	33
1.5 Méthodes de Monte Carlo par chaîne de Markov	37
2 Pistage multicible et Track-before-detect	44
2.1 Observations ponctuelles	46
2.1.1 Méthodes historiques	46
2.1.2 Cadre RFS	49
2.2 Observations brutes	54
2.2.1 Méthodes initiales par batch	55
2.2.2 Méthodes récursives	55
2.2.3 Sur données réelles	57
3 Etude de filtres particuliers multicibles	60
3.1 Filtre particulaire IPMCMC (Interacting Population-based Markov Chain Monte Carlo particle filter)	60
3.2 Filtre particulaire ATRAPP (Adaptive Target Resampling Auxiliary Parallel Partition particle filter)	67
3.3 Filtre particulaire avec transition markovienne auxiliaire et crossover	74

3.3.1	Matrice de transition auxiliaire	75
3.3.2	Densité de transition auxiliaire	78
4	Simulations et résultats	85
4.1	Avant-propos	85
4.2	Modèle de Swerling 3	87
4.3	Modèle de Swerling 0	92
5	Choix de modélisation du système	97
5.1	Modèle dynamique	97
5.2	Modèle d'observation	99
5.2.1	Traitement temporel (en distance) et effet Doppler	99
5.2.2	Traitement angulaire	108
5.3	Calcul de vraisemblance	114
5.3.1	Rappel du modèle d'observation	115
5.3.2	Modèles des paramètres phase et amplitude	115
5.3.3	Définition et expressions des vraisemblances	121
5.4	Gating (fenêtrage)	139
6	Mesures de performance multicible	150
6.1	Mesure par Schéma d'Affectation Optimale (Optimal Subpattern Assignment (OSPA) Metric)	150
6.2	Divergence de Kullback-Leibler	151
	Conclusion et perspectives	153
	Références	165
A	Particle filters with auxiliary Markov transition	166

Introduction

La poursuite de cibles est un domaine faisant appel à de nombreux concepts et méthodes de l'estimation, et s'appliquant à différents domaines, comme le maritime, le spatial, la biologie ou encore l'économie. Le cadre de l'estimation bayésienne permet de traiter ces problèmes d'estimation dont le but est de retrouver l'état d'un système (les cibles) à partir de données, bruitées et souvent partielles, récoltées à l'aide d'un ou plusieurs capteurs, et à partir d'une connaissance *a priori* du système. Les problèmes de poursuite multi-cibles comportent la détection, l'estimation de l'état des cibles (position, vitesse, ...) et le suivi de trajectoires, la détermination du nombre de cibles, l'identification de cibles, entre autres. Le suivi de trajectoires multi-cibles inclut une estimation successive des états des cibles dans le temps, introduisant un traitement séquentiel à mesure que des données sont récupérées par le ou les capteurs. Le but est d'exprimer la distribution *a posteriori* de l'état des cibles connaissant les observations récoltées jusque-là (grâce au modèle d'observation) et à partir d'une connaissance *a priori* (donnée par le modèle d'état): il s'agit de filtrage. Le filtrage peut être opéré en temps réel à l'arrivée d'une nouvelle observation si les temps de calcul le permettent.

Le cas le plus simple de poursuite de cibles est le cas où une seule cible est présente, est détectée, et génère un point d'observation (ou observation ponctuelle). Alors l'observation est associée à la cible, et l'estimation bayésienne fournit une estimation de l'état de la cible. Ce cas simple peut être enrichi si la cible est considérée comme pouvant être présente ou non (probabilité d'existence), détectée ou non (probabilité de détection), en considérant les fausses alarmes. Si de plus plusieurs cibles peuvent être présentes, il s'agit de poursuite multi-cibles, et la dimension de l'espace d'état augmente alors avec le nombre de cibles. Dans le cas d'observations ponctuelles (de détections), un ensemble de points d'observation est généré à chaque pas de temps, et des problèmes d'association des cibles aux observations se posent. De nombreuses méthodes tentent de répondre à ce problème, intégrant l'éventualité que les cibles ne soient pas toujours détectées, et les fausses alarmes. Seulement, pour que ces algorithmes puissent être mis en place, il faut que les cibles soient détectées régulièrement afin de réussir à les poursuivre. Cela n'est pas toujours le cas, notamment lorsque le [rapport signal sur bruit \(RSB\)](#)

est faible, et la phase de détection perd de l'information contenue dans les données. Il peut donc être nécessaire de traiter les données brutes (avant détection) afin de réussir à poursuivre certaines cibles en tirant profit de toute l'information contenue dans les données brutes. C'est dans ce contexte de suivi avant détection, ou en anglais [Track Before Detect \(TBD\)](#), que s'inscrit ce manuscrit. Dans le cas du radar, c'est donc l'image-radar brute qui peut être utilisée, plutôt que les points pouvant en être tirés. Le radar envoie une onde électromagnétique qui est réfléchi au contact d'un objet et revient sur le radar. Une image-radar est composée de cellules de résolution (en distance et en angle), et dans chacune de ces cellules est reporté un niveau de signal retour. Chaque cible contribue, selon son état et sa capacité de réflexion de l'onde, à l'image-radar. Le problème d'affectation entre les observations et les cibles n'existe plus, mais la façon dont les cibles contribuent ensemble à l'observation doit être modélisée. C'est ce modèle d'observation qui donnera accès à la fonction de vraisemblance.

Dans le cas où les modèles dynamiques et d'observation sont linéaires et gaussiens, le filtre de Kalman est la solution optimale du filtre récursif bayésien. Des variantes du filtre de Kalman (filtre de Kalman étendu, filtre de Kalman *unscented*) permettant en linéarisant ou en faisant des approximations gaussiennes de résoudre le problème de filtrage. Mais ces variantes sont limitées lorsque la non linéarité des modèles est trop forte et lorsque le bruit n'est pas additif gaussien. Alors souvent les équations de filtrage bayésien ne possèdent pas de solutions explicites, et il n'est pas possible d'accéder à la distribution *a posteriori* de l'état des cibles sachant les observations. Basée sur le principe de Monte Carlo, les filtres particuliers proposent une solution au problème de filtrage en représentant les distributions par un nuage de points pondérés appelés particules. Afin de générer ces particules, des stratégies d'échantillonnage sont mises en place, en essayant de créer de la diversité dans les particules et en limitant la variance des poids affectés aux particules. Les particules pondérées sont échantillonnées de manière récursive, les particules de l'instant précédent apportant une approximation de la distribution des états à l'état précédent et servant pour l'échantillonnage des nouvelles particules à l'instant présent.

Ainsi les algorithmes de suivi multi-cibles en contexte [TBD](#) (avec radar) doivent s'adapter au problème multi-cibles, intégrer le modèle d'observation du signal brut (non seuillé), qui est ici l'image-radar, et dans le cas de l'utilisation de filtres particuliers s'appuyer sur des méthodes d'échantillonnage adaptées et efficaces.

Parmi les méthodes d'échantillonnage, deux font l'objet d'analyse et de comparaison dans ce manuscrit: il s'agit des méthodes de Monte Carlo par chaîne de Markov et du filtre particulier auxiliaire. Ces deux principes sont respectivement mis en œuvre dans le filtre particulier [Interacting Population-based Monte Carlo Markov Chain \(IPMCMC\)](#) et dans le filtre particulier [Adaptive Target Resampling Auxiliary Parallel Partition \(ATRAPP\)](#). Ces deux algorithmes proposent une solution particulière à la résolution des équations du filtrage bayésien appliqué au problème multi-cibles. Puisqu'il est bien souvent impossible d'échantillonner directement selon

la distribution *a posteriori* désirée, ni selon une distribution définie comme optimale (selon le critère d'une variance réduite des poids des particules), les stratégies d'échantillonnage citées ci-avant permettent d'échantillonner les particules selon une distribution à partir de laquelle il est possible de tirer les particules multi-cibles. Les méthodes de Monte Carlo par chaîne de Markov génèrent les échantillons via une chaîne de Markov dont la distribution stationnaire est la distribution multi-cibles *a posteriori*. Les échantillons sont tirés successivement à l'aide d'un noyau de Markov, et les particules sont conservées après une période suffisante de simulation (période de *burn-in*) permettant de garantir la convergence vers la distribution stationnaire. Plusieurs chaînes de Markov peuvent être lancées simultanément afin de réaliser des calculs en parallèle et ainsi gagner du temps d'exécution. Ces stratégies permettent, à l'aide d'un choix judicieux des noyaux de Markov, d'explorer l'espace d'état et de générer des particules qui ont le même poids. Le filtre particulaire auxiliaire fait appel à une variable auxiliaire qui va permettre d'orienter la sélection des particules avant que ces dernières ne soient propagées. L'idée est de pré-sélectionner les particules ancêtres prometteuses. Les poids auxiliaires, permettant d'échantillonner la variable auxiliaire, font bien souvent intervenir l'observation. Ainsi la sélection préalable des particules est orientée par l'observation: l'information contenue dans l'observation est utilisée pour orienter le tirage des particules, et ce avant la pondération, afin d'imiter le tirage via la distribution optimale.

Dans un contexte multi-cibles, il est courant de prendre pour hypothèse l'indépendance dynamique des cibles: les cibles se déplacent indépendamment les unes des autres. Cette hypothèse est faite dans le reste du manuscrit et dans les deux algorithmes présentés précédemment. Ainsi lors d'une simulation du modèle dynamique, les cibles sont simulées les unes indépendamment des autres. En plus de cela, de nombreux filtres particulaires multi-cibles, comme le filtre [ATRAPP](#), font l'hypothèse de l'indépendance postérieure des cibles. Cela consiste à considérer après chaque pas de temps que la distribution multi-cibles *a posteriori*, obtenue à l'aide de la représentation particulaire, peut être factorisée sur les cibles. Bien sûr, cette hypothèse ne peut être vraie dans un contexte multi-cible dans lequel les cibles sont proches, et dans lequel leurs contributions à l'observation se superposent. En revanche cette hypothèse, permet de faciliter la mise en œuvre d'algorithmes. Il devient alors possible, comme dans le filtre particulaire [ATRAPP](#), de réaliser un traitement cible par cible, pour l'échantillonnage notamment. Les deux algorithmes [IPMCMC](#) et [ATRAPP](#) construisent de nouvelles particules dont chacune des cibles peut venir d'une particule précédente différente: cela s'apparente à du *crossover*, une hybridation entre les particules, et génère de la diversité dans les particules proposées. Ce brassage des particules permet une bonne exploration de l'espace d'état (de grande dimension, d'autant plus grande qu'il y a de cibles) en espérant construire ainsi des particules multi-cibles bien représentatives.

Pour s'affranchir de l'hypothèse d'indépendance postérieure des cibles, qui est fautive dans

de nombreux cas, tout en conservant l'idée de variable auxiliaire et le principe de *crossover*, une nouvelle méthode, introduite dans ce manuscrit, s'appuie sur une transition markovienne auxiliaire. Des poids auxiliaires sont déterminés à la manière d'un filtre particulaire auxiliaire. Dans un premier temps est introduite la matrice de transition auxiliaire qui permet de générer le *crossover* en sélectionnant une particule différente pour chacune cibles à partir desquelles elles seront échantillonnées. Cette matrice de transition associe l'indice d'une particule ancêtre à une combinaison d'indices des différentes particules à partir desquelles seront échantillonnées chacune des cibles. Les particules sont ensuite pondérées. Les poids auxiliaires et la matrice de transition auxiliaires constituent des paramètres de conception qu'il faut choisir. Ils peuvent être déterminés à l'aide du critère d'optimalité qu'est la minimisation de la variance des poids. Seule la recherche exhaustive permet d'arriver aux combinaisons d'indices atteignant ce minimum. Une autre solution est alors d'introduire une densité de transition auxiliaire: la matrice est remplacée par une densité qui au lieu de proposer de piocher parmi une recombinaison des particules existantes, laisse le choix parcourir tout l'espace d'état. De même cette densité est un paramètre de conception de l'algorithme, et le minimum de la variance est réduit par rapport au choix d'une matrice de transition auxiliaire. Il est vérifié que ces deux options peuvent être vues comme des généralisations du filtre particulaire auxiliaire classique.

Remarque – De nombreux termes anglais et leurs acronymes sont présents tout au long de ce manuscrit, la majorité de la littérature du domaine étant en langue anglaise. Un effort a été fait pour traduire ces termes lorsque cela était possible et pertinent. Le choix a parfois été fait de garder les termes anglais et leurs acronymes lorsqu'ils sont considérés comme incontournables ou répandus.

Organisation du manuscrit

Ce manuscrit couvre plusieurs thématiques nécessaires à la mise en place d'algorithmes de poursuite multi-cibles dans un contexte TBD, qui sont présentées selon la structure suivante:

- Le chapitre 1 introduit les notations et le principe de l'estimation bayésienne récursive (ou filtrage bayésien), permettant de déterminer, à partir des modèles dynamique et d'observation, la distribution de l'état connaissant les observations récoltées jusque là, et de manière récursive. C'est à partir de cette distribution que peuvent être extraites des estimations de l'état avec leurs incertitudes. Sont rappelées les solutions d'implémentation de type filtre de Kalman (et ses dérivées), et sont plus longuement développées les solutions issues des méthodes de Monte Carlo, notamment les méthodes de Monte Carlo séquentielles, ou filtrage particulaire, et en anglais [sequential Monte Carlo \(SMC\)](#). Ainsi différents filtres particuliers (auxiliaire, *Rao-Blackwellisé*) sont introduits, ainsi que les

méthodes [Monte Carlo par chaînes de Markov \(MCMC\)](#).

- Le chapitre [2](#) présente un état de l’art de la poursuite de cibles, tout d’abord dans le cadre d’observations dites ponctuelles (ou seuillées, dans la section [2.1](#)), puis avec des données brutes (non seuillées, dans la section [2.2](#)). Travailler avec des données brutes permet de définir l’approche appelée [TBD](#): les cibles sont suivies (pistées) avant d’être détectées (c’est-à-dire d’être représentée par une observation seuillée).
- Le chapitre [3](#) présente l’étude d’algorithmes permettant à l’aide de méthodes [SMC](#) d’approcher la distribution multi-cibles d’intérêt. Des algorithmes récents et complexes, ayant été développés pour répondre spécifiquement à la problématique multi-cibles en contexte [TBD](#), et aux méthodes diverses de rééchantillonnage ont été sélectionnés et sont développés et analysés en sections [3.1](#) et [3.2](#).
 - La section [3.1](#) présente le filtre particulaire [IPMCMC](#) [[BDB12](#)] qui repose sur l’utilisation de méthodes de [MCMC](#), appliqués au problème multi-cibles grâce à un noyau de transition spécifique.
 - La section [3.2](#) présente le filtre particulaire [ATRAPP](#) [[UMGFG17](#)] qui repose sur une hypothèse d’indépendance des cibles (*a posteriori*) permettant de proposer un schéma de filtrage particulaire auxiliaire adaptatif.

Ces deux algorithmes basés sur une approche particulaire pour résoudre le problème d’estimation séquentielle, utilisent deux méthodes bien distinctes afin de générer les particules, et chacun introduit cependant à sa manière un brassage des particules.

Partant du constat que l’hypothèse d’indépendance des cibles (*a posteriori*) ne peut être toujours valable,

- la section [3.3](#) introduit au filtre particulaire la notion de transition markovienne auxiliaire. Ces transitions se déclinent sous forme de matrice de transition auxiliaire ou de densité de transition auxiliaire, permettant ainsi lors de l’échantillonnage un brassage des particules (les états des cibles d’une nouvelle particule peuvent venir de particules différentes), générant du *crossover*, et amenant une diversité des nouvelles particules échantillonnées.
- Le chapitre [4](#) met en œuvre les deux premiers algorithmes du chapitre [3](#) avec des simulations numériques sur des scénarios multi-cibles afin d’éprouver les deux méthodes.
- Le chapitre [5](#) détaille le modèle dynamique des cibles choisi, ainsi que le modèle d’observation pour le radar dans un contexte [TBD](#). Le modèle d’observation est décrit par la fonction équipement du radar et l’amplitude du signal, et donne à partir de l’état des cibles une

image radar complexe. Les modèles de Swerling déterminant la fluctuation de l'amplitude des cibles sont introduits. Plusieurs fonctions de vraisemblance sont données selon le cas étudié (selon le modèle de Swerling, si le travail est réalisé sur les données complexes ou non, s'il n'y a qu'une cible ou plusieurs,...). Enfin un fenêtrage (ou *gating*) est proposé afin de réduire la dimension de l'observation, en s'adaptant au cadre TBD et au traitement du filtre particulière.

- Le chapitre 6 présente deux mesures d'analyse de la performance d'algorithme de la poursuite multi-cibles.
- L'annexe A, rédigée en anglais, présente une étude approfondie de l'algorithme de filtrage particulière avec transitions markoviennes auxiliaires déjà présenté dans la partie 3.3.

Publications

Articles en conférence

- Cuillery, A., et Le Gland, F. Track-before-detect on radar image observation with an adaptive auxiliary particle filter. In *2019 22th International Conference on Information Fusion (FUSION)*. IEEE, 2019
- Cuillery, A., et Le Gland, F. Particle filters with auxiliary markov transition. application to crossover and to multitarget tracking. In *2021 24th International Conference on Information Fusion (FUSION)*. IEEE, 2021

Pré-publication

- Cuillery, A., et Le Gland, F. Particle filters with auxiliary Markov transition. Application to crossover and to multitarget tracking. Publication Hal hal-03467603, 2021

Chapitre 1

Filtrage bayésien et solutions d'implémentation

1.1 Modélisation et filtrage bayésien optimal

Le but du filtrage bayésien, issu de l'étude du filtrage optimal et stochastique [AM79, Si3], est d'estimer l'état inconnu d'un système qui évolue dans le temps, observé indirectement grâce à des mesures bruitées. Le **système**, ou **cible**, peut être un avion, une personne, une cellule biologique, etc. L'état du système fait référence à une collection de paramètres qui décrivent complètement ce système, comme par exemple la position, la vitesse, l'accélération, l'intensité, la présence ou l'absence, etc. Les **observations**, ou **mesures**, obtenues à partir de la cible sont par exemple des mesures radar ou bien des images vidéo. Le cadre bayésien permet de décrire ce système et la façon dont nous l'observons en intégrant des incertitudes sur l'évolution et l'observation via des modèles probabilistes. Ces incertitudes sont introduites par des bruits qui viennent perturber les modèles déterministes dynamiques (évolution du système) et d'observation (information sur le système, mesures). Bien que la dynamique du système ne soit pas vraiment stochastique, le bruit d'état vient décrire l'incertitude sur le modèle utilisé (et donc choisi) pour modéliser l'évolution temporelle de l'objet dont on est censé ignorer l'état et le comportement. Le bruit d'observation quant à lui traduit une erreur de mesure.

Le temps est discret et indicé: $t \in \mathbb{T} = \{0, \dots, T\}$ avec $T \in \mathbb{N}$. L'état de la cible et les observations sont considérés comme réalisations de vecteurs aléatoires. On va estimer récursivement (dans le temps) l'état inconnu du système à partir des observations bruitées en s'appuyant sur une connaissance *a priori* de la dynamique du système ainsi que sur l'information incertaine

donnée par les observations.

Modèle dynamique

On considère un processus $\{X_t\}_{t \geq 0}$, qui à chaque instant t

- est défini par un vecteur d'état $x_t \in \mathcal{X} \subseteq \mathbb{R}^{d_x}$
- est décrit par une **fonction de transition** de Markov du premier ordre, c'est-à-dire que l'état actuel x_t ne dépend que de l'état immédiatement précédent x_{t-1}

$$x_t = m_t(x_{t-1}, v_t) \quad (1.1)$$

- m_t : fonction de transition (modèle dynamique) éventuellement non linéaire
 v_t : bruit de dynamique aléatoire de loi connue
 les $\{v_t\}_{t \geq 0}$ sont indépendants et identiquement distribués (i.i.d.)

En conséquence, la transition se détermine par une **densité de transition** de Markov, induite par le bruit v_t

$$X_t \sim f_t(\cdot | x_{t-1}) \quad (1.2)$$

en précisant $\Pr(X_t \in dx | X_{t-1} = x_{t-1}) = f_t(x | x_{t-1})dx$.

Il est nécessaire pour caractériser complètement cette séquence de Markov de définir une densité initiale $\mu_0(\cdot)$ de l'état caché x_0 au temps $t = 0$

$$X_0 \sim \mu_0(\cdot) \quad (1.3)$$

en précisant $\Pr(X_0 \in dx) = \mu_0(x)dx$.

On définit alors la trajectoire de la cible à l'instant t par $x_{0:t} = \{x_0, \dots, x_t\}$.

Modèle d'observation

On considère un processus $\{Z_t\}_{t \geq 0}$, qui à chaque instant t

- est défini par un vecteur d'observation $z_t \in \mathcal{Z} \subseteq \mathbb{R}^{d_z}$
- est décrit par une **fonction d'observation**

$$z_t = h_t(x_t, w_t)$$

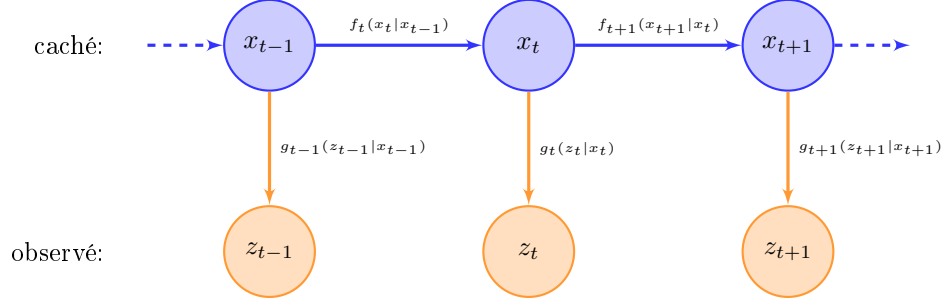


Figure 1.1: Processus bayésien du système

h_t : fonction d'observation (modèle de mesure) éventuellement non linéaire
 w_t : bruit d'observation aléatoire de loi connue
 les $\{w_t\}_{t \geq 0}$ sont indépendants et identiquement distribués (i.i.d.)

On suppose également que les bruits d'observations sont indépendants de la suite des états cachés. Une condition suffisante est que la condition initiale, les bruits de dynamique et les bruits d'observation soient mutuellement indépendants. En particulier, le bruit d'observation w_t est indépendant de l'état caché x_t . En conséquence la modélisation de l'observation se définit par une densité de probabilité conditionnelle, induite par le bruit w_t

$$Z_t \sim g_t(\cdot|x_t) \quad (1.4)$$

en précisant $\Pr(Z_t \in dz|X_t = x_t) = g_t(z|x_t)dz$. Cette densité, vue comme fonction de la variable x_t , est appelée fonction de **vraisemblance** (ou simplement vraisemblance).

L'historique des mesures est $z_{0:t} = \{z_0, \dots, z_t\}$.

Les fonctions m_t et h_t , ainsi que les densités f_t et g_t , sont indicées par le temps et peuvent être de manière plus simple considérées comme homogènes (indépendantes du temps), ce qui sera le cas par la suite afin de simplifier les notations. Cette modélisation s'inscrit dans les modèles dits modèles de Markov cachés (Hidden Markov Models, HMM). Par "modèles de Markov" on qualifie la transition du système, et par "cachés" le fait que l'on ne connaisse pas l'état (puisque qu'on cherche à l'estimer) et que l'on n'ait accès à de l'information à son sujet que par les observations. Le réseau bayésien tel que décrit précédemment est illustré sur la figure 1.1.

Les équations (1.2), (1.3) et (1.4) définissent complètement le modèle bayésien, dans lequel

(1.2) et (1.3) définissent une loi de probabilité sur les états joints du processus (ou trajectoire),

$$p(x_{0:T}) = \mu(x_0) \prod_{t=1}^T f(x_t | x_{t-1}) \quad (1.5)$$

tandis que (1.4) permet de définir la vraisemblance de la trajectoire conditionnellement à l'historique des observations,

$$p(z_{0:T} | x_{0:T}) = \prod_{t=0}^T g(z_t | x_t) \quad (1.6)$$

Grâce aux deux équations (1.5) et (1.6), on va chercher à extraire une inférence sur notre processus, c'est-à-dire déterminer une loi de probabilité sur notre trajectoire conditionnellement à l'historique des observations qui représente l'information que l'on a. Il s'agit donc de trouver au choix à chaque instant t

- la densité de probabilité sur la trajectoire $p(x_{0:t} | z_{0:t})$
- la densité de probabilité marginale $p(x_t | z_{0:t})$

Ces densités sont aussi référées comme étant les densités *a posteriori*, et les deux sont appelées indifféremment **densité de filtrage**. Dans la suite on s'intéressera donc uniquement à la loi marginale $p(x_t | z_{0:t})$, et dans l'idée du filtrage séquentiel il est souhaitable de l'exprimer récursivement à chaque instant en fonction de $p(x_{t-1} | z_{0:t-1})$, avec comme point de départ (ou première "prédiction") $p(x_0 | z_\emptyset) = \mu(x_0)$, avec $z_\emptyset$ représentant l'absence d'observation lors de l'initialisation. La figure 1.2 illustre le principe de récursivité.

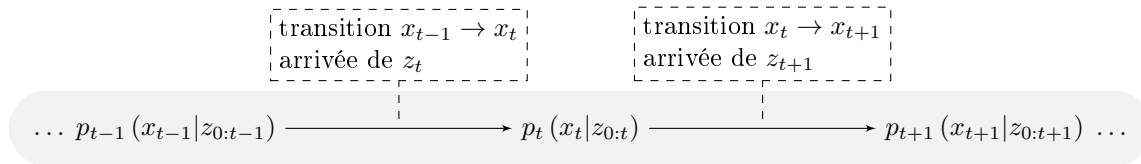


Figure 1.2: Densité de filtrage séquentielle

Le filtre récursif bayésien, qui constitue une solution optimale au problème formulé ci-dessus, est classiquement donné sous la forme de deux étapes:

1. Prédiction:

- Elle définit la densité de prédiction $p(x_t | z_{0:t-1})$
- à partir de la densité de filtrage au temps précédent $p(x_{t-1} | z_{0:t-1})$ et de la densité de transition $f(x_t | x_{t-1})$.

$$p(x_t|z_{0:t-1}) = \int_{\mathcal{X}} f(x_t|x_{t-1})p(x_{t-1}|z_{0:t-1})dx_{t-1} \quad (1.7)$$

2. Mise à jour (ou Update):

- Elle définit la densité de de filtrage $p(x_t|z_{0:t})$
- à partir de la densité de prédiction $p(x_t|z_{0:t-1})$ et de la vraisemblance $g(z_t|x_t)$.

$$p(x_t|z_{0:t}) = \frac{g(z_t|x_t)p(x_t|z_{0:t-1})}{\int_{\mathcal{X}} g(z_t|x)p(x|z_{0:t-1})dx} \quad (1.8)$$

La prédiction (1.7) traduit bien l'incertitude donnée par le modèle dynamique sur le déplacement et l'évolution de l'objet au nouveau pas de temps t , tandis que la mise à jour (1.8) vient intégrer l'information apportée par la nouvelle mesure z_t . La figure 1.3 illustre la récursivité en deux étapes.

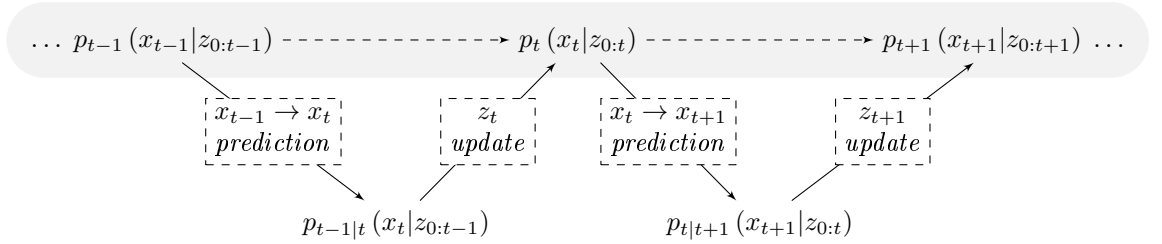


Figure 1.3: Filtrage de Bayes séquentiel

Une fois la densité de filtrage établie $p(x_t|z_{0:t})$, on souhaite estimer x_t . Les deux estimateurs les plus classiques sont l'estimateur de maximum a posteriori (MAP) et l'estimateur par minimum d'erreur quadratique moyenne, qui revient à calculer l'espérance a posteriori (EAP), donnés par

- MAP: estimateur donnant l'état x_t maximisant la densité de filtrage

$$x_{t,MAP} = \underset{x_t}{\operatorname{argmax}} p(x_t|z_{0:t})$$

- EAP: estimateur donnant la moyenne de l'état x_t

$$x_{t,EAP} = \mathbb{E}[x_t|z_{0:t}] = \int x_t p(x_t|z_{0:t}) dx_t$$

Selon la forme choisie pour les densités intervenant dans chacune de ces équations, il n'est pas toujours possible d'en obtenir des expressions exactes, à part dans certains cas spécifiques, comme les modèles linéaires gaussiens, ou le cas d'un espace d'état discret et fini. Dans le cas des modèles linéaires gaussiens, c'est-à-dire lorsque les fonctions du modèle dynamique m_t ainsi que d'observation h_t sont linéaires et que les bruits sont gaussiens et additifs, la traduction de ces équations dans ce cadre est le filtre de Kalman. Les autres cas vont faire appel à des techniques alternatives.

Les méthodes dites à grille proposent comme solution approchée une discrétisation de l'espace d'état sur laquelle est calculée la densité en chaque point du maillage déterministe, ce qui se traduit par une somme de fonctions de Dirac. Ces méthodes peuvent être envisagées dans des problèmes à faible dimension puisque la complexité augmente exponentiellement avec la dimension de l'espace d'état et les ressources requises sont alors importantes. Le filtre de Kalman étendu (Extended Kalman Filter, EKF), le filtre de Kalman *unscented* (Unscented Kalman Filter, UKF) et le filtre à somme de gaussiennes (Gaussian Mixture, GM), sont des variantes du filtre de Kalman pour les problèmes non linéaires. L'EKF va faire appel à une approximation du premier ou deuxième ordre de la moyenne et de la covariance en linéarisant localement les fonctions m_t et h_t . Le filtre de Kalman *unscented* utilise quand à lui des points paramétrisant la moyenne et la covariance (ici c'est la gaussienne qui est approximée) et auxquels sont appliquées les transformations non linéaires. Lorsque la densité a posteriori est représentée par une somme de gaussiennes, ou aussi appelée mélange de gaussiennes, chaque composante de la somme est une gaussienne propagée selon un filtre de Kalman et ses dérivés. Le nombre de composantes formant ce mélange augmente exponentiellement avec le temps. Les méthodes par approche de Monte Carlo sont détaillées ci-dessous.

1.2 Principe de Monte Carlo

Le principe de Monte Carlo est central au développement des techniques d'implémentation présentées ci-après et est donc développé dans un premier temps dans cette section de façon générale.

Considérons une variable aléatoire X de densité de probabilité p . Le principe de Monte Carlo repose sur l'idée qu'il est possible d'approcher la densité de probabilité p , par un ensemble d'échantillons indépendants et identiquement distribués (i.i.d.), appelés également **particules**, et noté $\{x^i\}_{i=1}^N$. Ces particules sont des réalisations de la variable aléatoire X . Il s'agit d'une représentation ponctuelle ou discrète, à caractère aléatoire puisqu'on échantillonne les particules selon une distribution. La densité p de X est d'autant mieux représentée que l'on possède un grand nombre N de particules.

Considérons l'échantillonnage de Monte Carlo "parfait" pour la densité de probabilité p , c'est-à-dire que l'on suppose avoir accès à un ensemble de N échantillons tirés exactement selon la densité p constituant nos particules

$$x^1, \dots, x^N \sim p(x)$$

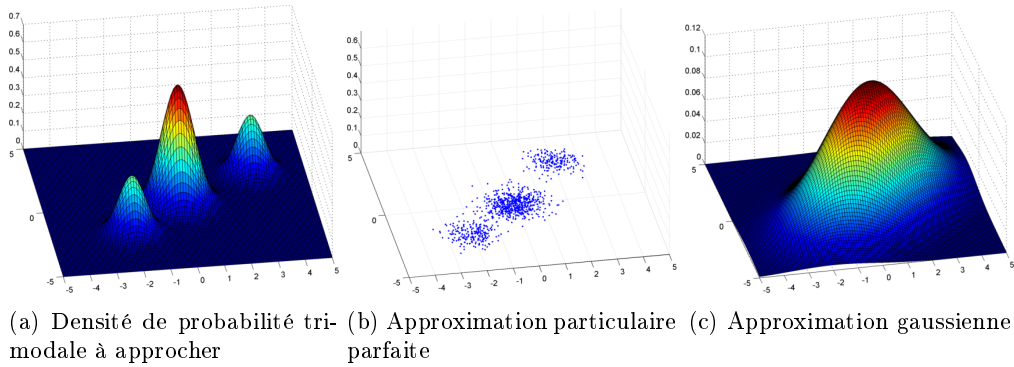


Figure 1.4: Densité de probabilité tri-modale et ses approximations

L'approximation de la densité p est alors donnée par la mesure empirique ponctuelle constituée de N mesures de Dirac équipondérées s'appuyant sur les $\{x^i\}_{i=1}^N$

$$p(x) \approx p_N(x) = \frac{1}{N} \sum_{i=1}^N \delta_{x^i}(x) \quad (1.9)$$

Les méthodes de Monte Carlo permettent d'approcher numériquement des quantités qui n'ont pas d'expressions analytiques explicites, notamment les intégrales, représentant de manière équivalente des espérances mathématiques

$$I(f) = \mathbb{E}_p[f(X)] = \int_{\mathcal{X}} f(x)p(x)dx \quad (1.10)$$

avec f une fonction p -mesurable et bornée. Ces intégrales apparaissent dans les équations de filtre bayésien, lors de marginalisation ou normalisation des densités étudiées. En utilisant l'approximation de p donnée par (1.9) dans l'expression de l'intégrale (1.10), l'approximation (aussi appelée estimation) suivante est obtenue

$$I(f) \approx I_N(f) = \int_{\mathcal{X}} f(x)p_N(x)dx = \frac{1}{N} \sum_{i=1}^N f(x^i) \quad (1.11)$$

L'estimation $I_N(f)$ de $I(f)$ est non biaisée. La loi forte des grands nombres assure que

l'approximation (1.11) converge presque sûrement (p.s.) vers l'expression (1.10) lorsque le nombre d'échantillons tend vers l'infini

$$I_N \xrightarrow[N \rightarrow \infty]{\text{p.s.}} I$$

Si $f(X)$ est à variance finie, c'est-à-dire

$$\text{Var}(f(X)) = \int_{\mathcal{X}} f^2(x)p(x)dx - I^2(f) < \infty$$

la variance de l'estimation est donnée par

$$\begin{aligned} \text{Var}(I_N(f)) &= \frac{1}{N} \left(\int_{\mathcal{X}} f^2(x)p(x)dx - I^2(f) \right) \\ &= \frac{\text{Var}(f(X))}{N} \end{aligned}$$

Le théorème centrale limite donne la convergence en distribution de l'erreur d'estimation

$$\sqrt{N} (I_N(f) - I(f)) \xrightarrow[N \rightarrow \infty]{\mathcal{L}} \mathcal{N}(0, \text{Var}(f(X)))$$

L'avantage des méthodes de Monte Carlo réside dans le fait que la variance de l'erreur d'approximation décroît à un taux $\mathcal{O}(1/N)$ qui est donc indépendant de la dimension de l'intégrale sur l'espace $\mathcal{X}$, contrairement à d'autres méthodes d'intégration numérique déterministes. Aussi, contrairement à des approximations discrètes sur des points de grilles déterministes, par exemple un ensemble de points $\{x^i\}_{i=1}^N$ répartis à égale distance sur l'espace $\mathcal{X}$, on peut concentrer les ressources des calculs dans les zones d'intérêt, c'est-à-dire les régions à forte densité de probabilité. En revanche, il n'est pas toujours trivial, voire souvent impossible, d'échantillonner selon la densité souhaitée p . En effet p peut être complexe à grande dimension, ou même connue qu'à une constante multiplicative près, c'est-à-dire on connaît seule l'expression $\tilde{p}(x)$ telle que $p(x) \propto \tilde{p}(x)$. Pour pallier cela, des techniques telles que l'échantillonnage par méthode de rejet, l'échantillonnage d'importance, ou méthodes par chaîne de Markov peuvent être employées.

1.3 Echantillonnage d'importance

L'échantillonnage d'importance (Importance Sampling, IS) est une méthode permettant d'approximer une intégrale ou une espérance par rapport à une densité p selon laquelle on ne sait pas directement échantillonner. Cette méthode requiert la connaissance a minima de l'expression de la densité à une constante multiplicative près, c'est-à-dire on connaît $\tilde{p}$ telle que $p(x) = \frac{\tilde{p}(x)}{\int \tilde{p}(x)}$.

L'idée est la suivante: des échantillons sont tirés à partir d'une distribution selon laquelle il est facile (ou en tout cas possible) d'échantillonner, qui est appelée *densité de proposition* ou *densité d'importance* et notée q . Cela permet d'obtenir une approximation particulière de cette densité. La densité q est choisie telle que le support de p soit inclus dans le support de q , c'est-à-dire $p(x) > 0 \Rightarrow q(x) > 0$. Cette densité est introduite dans l'expression cherchée et fait apparaître alors un terme correcteur comme suit

$$I(f) = \mathbb{E}_p[f(x)] = \int_{\mathcal{X}} f(x) \frac{p(x)}{q(x)} q(x) dx = \mathbb{E}_q \left[f(x) \frac{p(x)}{q(x)} \right] \quad (1.12)$$

avec l'apparition du terme correcteur $p(x)/q(x)$. En échantillonnant selon q , une approximation particulière va être déduite pour cette dernière

$$q(x) \approx q_N(x) = \frac{1}{N} \sum_{i=1}^N \delta_{x^i}(x) \quad (1.13)$$

Deux cas se présentent alors:

1. L'expression exacte de p est connue

L'introduction de l'approximation q_N de q permet d'obtenir directement une approximation de la densité d'intérêt p ainsi qu'en conséquence de l'intégrale souhaitée.

$$p(x) \approx p_{N,IS}(x) = \frac{1}{N} \sum_{i=1}^N \tilde{w}(x^i) \delta_{x^i}(x) \quad (1.14)$$

avec

$$\tilde{w}(x^i) = \frac{p(x^i)}{q(x^i)}$$

les $\tilde{w}(x^i)$ étant dénommés "poids d'importance". En intégrant l'approximation de q de (1.13) dans (1.12), ou de façon équivalente l'approximation par importance de p (1.14) dans (1.10)

$$I(f) \approx I_{N,IS}(f) = \frac{1}{N} \sum_{i=1}^N \tilde{w}(x^i) f(x^i)$$

Cet estimateur $I_{N,IS}(f)$ est sans biais et a pour variance

$$\text{Var}(I_{N,IS}(f)) = \frac{1}{N} \text{Var} \left(\frac{p}{q} f(X) \right)$$

2. L'expression de p n'est connue qu'à une constante multiplicative près

$$p(x) = \frac{\tilde{p}(x)}{\int \tilde{p}(x)} \quad (1.15)$$

L'approximation de q (1.13) est utilisée pour approcher le numérateur et le dénominateur de l'expression (1.15) de p

$$p(x) \approx p_{N,IS}(x) = \frac{\frac{1}{N} \sum_{i=1}^N \tilde{w}(x^i) \delta_{x^i}(x)}{\frac{1}{N} \sum_{i=1}^N \tilde{w}(x^i)} = \sum_{i=1}^N w(x^i) \delta_{x^i}(x)$$

avec cette fois

$$\tilde{w}(x^i) = \frac{\tilde{p}(x^i)}{q(x^i)} \quad \text{et} \quad w(x^i) = \frac{\tilde{w}(x^i)}{\sum_{i=1}^N \tilde{w}(x^i)}$$

ce qui permet d'écrire

$$I(f) \approx I_{N,IS}(f) = \sum_{i=1}^N w(x^i) f(x^i)$$

Cet estimateur $I_{N,IS}$ auto-normalisé est biaisé (avec un biais d'ordre $1/N$) et à une variance d'ordre $1/N$ (mais l'expression exacte de cette variance n'est pas disponible) [Kon92].

Par exemple, la densité de probabilité tri-modale de la figure 1.4a est reprise et l'on suppose que l'on ne sait pas échantillonner selon cette densité mais que l'on connaît son expression (à une constante multiplicative près). En faisant le choix d'une densité de proposition gaussienne selon laquelle on sait échantillonner, les résultats de la pondération des particules obtenus sont présentés sur la figure 1.5.

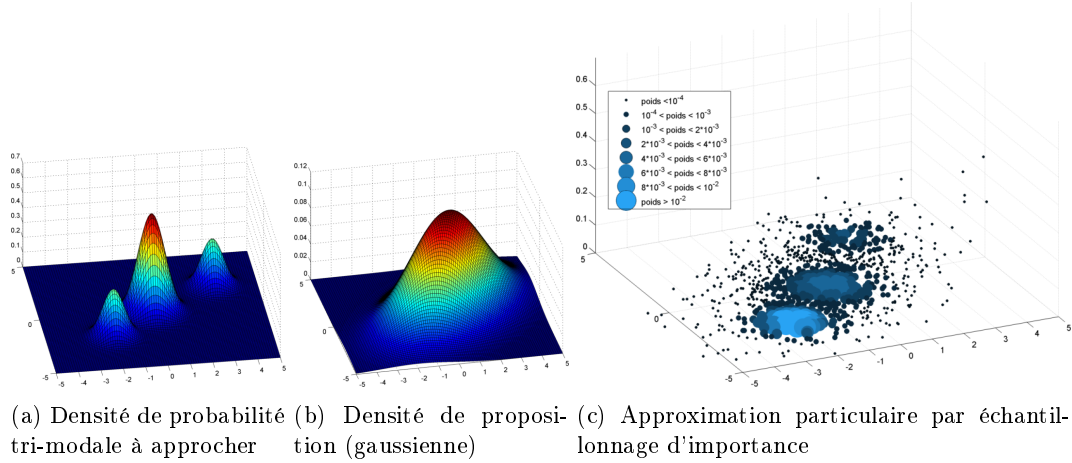


Figure 1.5: Densité et particules pondérées générées par échantillonnage d'importance

1.4 Filtrage particulaire

Dans les cas où le système étudié présente des non linéarités ou ne suit pas une statistique gaussienne, il n'est pas possible d'obtenir des solutions explicites des équations pour les problématiques d'inférence bayésienne. Une approximation dite particulaire du filtre de Bayes peut être proposée, comme introduite initialement par [GSS93]. Il s'agit d'une approche par la méthode de Monte Carlo, propagée au cours du temps donc dite séquentielle. Les termes de Monte Carlo Séquentiel (ou Sequential Monte Carlo, SMC) et de Filtrage Particulaire (ou Particle Filter, PF) seront utilisés indifféremment. De nombreuses références existent sur le sujet, parmi lesquelles [GSS93, RAG04, AMGC02, DJ11]. Ces méthodes présentent deux intérêts majeurs :

- aucune restriction n'est imposée sur les modèles mathématiques utilisés pour décrire le système;
- la convergence est garantie sous réserve de ressources de calcul suffisantes.

Ainsi, il n'est pas nécessaire de faire appel à des techniques de linéarisation ou des approximations analytiques de fonctions, contrairement par exemple au filtre de Kalman étendu. En contre partie, les méthodes SMC sont très coûteuses en termes de puissance de calcul.

1.4.1 Filtre particulaire et échantillonnage d'importance séquentiel

Il s'agit désormais d'utiliser la formulation séquentielle du problème de filtrage afin de pouvoir appliquer une technique de Monte Carlo séquentiellement. A chaque pas de temps, la densité

de filtrage est donc approximée par méthode de Monte Carlo avec un nuage de particules pondérées, comme illustré sur la figure 1.6. Les poids des particules sont ici notés $w_t^i = w(x_t^i)$.

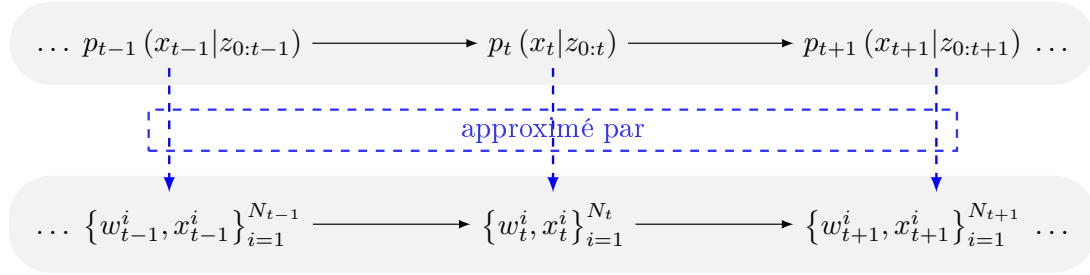


Figure 1.6: Schéma du filtre particulaire

La densité de filtrage à l'instant $t-1$ est approximée par un ensemble de particules pondérées

$$p_{t-1}(x_{t-1}|z_{0:T-1}) \approx \sum_{i=1}^N w_{t-1}^i \delta_{x_{t-1}^i}(x_{t-1}) \quad (1.16)$$

La version "classique" du filtre particulaire s'appuie sur l'échantillonnage d'importance décrit ci-avant.

La densité de filtrage au temps t est approchée par une étape d'échantillonnage d'importance s'appuyant sur son expression séquentielle exprimée par (1.7) et (1.8), l'approximation (1.16) et une densité de proposition $q(x_t|x_{t-1}^i, z_t)$ dont le support contient celui de $p_t(x_t|z_{0:t})$. L'approximation de la densité de filtrage est alors donnée par le nouvel ensemble de particules,

$$x_t^1, \dots, x_t^{N_t} \sim q(x_t|x_{t-1}^i, z_t)$$

pondérées via le filtre particulaire,

$$p_t(x_t|z_{0:t}) \approx \sum_{i=1}^N w_t^i \delta_{x_t^i}(x_t)$$

avec

$$w_t^i = \frac{\tilde{w}_t^i}{\sum_{i=1}^N \tilde{w}_t^i} \quad \text{et} \quad \tilde{w}_t^i = w_{t-1}^i \frac{g(z_t|x_t^i) f(x_t^i|x_{t-1}^i)}{q(x_t^i|x_{t-1}^i, z_t)} \quad (1.17)$$

Cette version classique du filtre particulaire par échantillonnage d'importance séquentiel développée dans l'algorithme 1 présente l'inconvénient majeur de la dégénérescence des poids [CMR05, Chapitre 7], correspondant à l'augmentation de la variance sur les poids des particules. La densité de proposition telle que présentée ci-dessus, $q(x_t|x_{t-1}^i, z_t)$, suggère par le condition-

Algorithme 1 : Filtre particulaire au temps t (échantillonnage d'importance séquentiel)

Input : $\{w_{t-1}^i, x_{t-1}^i\}_{i=1}^N$ et une nouvelle observation z_t
Output : $\{w_t^i, x_t^i\}_{i=1}^N$

```

1. Prédiction
for  $i = 1$  to  $N$  do
    | Echantillonner  $x_t^i \sim q(\cdot | x_{t-1}^i, z_t)$ 
end
2. Mise à jour
for  $i = 1$  to  $N$  do
    | Calculer le poids non normalisé  $\tilde{w}_t^i = w_{t-1}^i \frac{g(z_t | x_t^i) f(x_t^i | x_{t-1}^i)}{q(x_t^i | x_{t-1}^i, z_t)}$ 
end
3. Normalisation
for  $i = 1$  to  $N$  do
    | Normaliser le poids  $w_t^i = \frac{\tilde{w}_t^i}{\sum_{i=1}^N \tilde{w}_t^i}$ 
end

```

nement en x_{t-1}^i que chaque particule i va être à son tour propagée, donnant naissance à x_t^i , et ce quelque soit son poids, même faible. Cette simulation séquentielle va voir les particules se disperser, et seules celles représentant suffisamment bien la densité, c'est-à-dire étant proche de l'état estimé, auront un poids significatif, les autres continueront à être propagées avec un poids proche de zéro. Afin d'éviter la propagation de particules qui ne représenteraient plus la densité ciblée, des techniques de rééchantillonnage ont été mises en place, permettant de recréer une population de particules représentatives. L'idée est de répliquer les particules à fortes pondération et d'abandonner celles avec un poids non représentatif. Il existe plusieurs stratégies de rééchantillonnage [DCM05], dont (nommé du plus au moins performant): le rééchantillonnage systématique, le stratifié, le résiduel, le multinomial. Les cartes ainsi redistribuées, le filtrage peut continuer. Cependant, en contre partie, en dupliquant certaines particules la diversité dans la population est réduite. Afin de n'appliquer le rééchantillonnage que lorsqu'il est nécessaire, c'est-à-dire lorsque des écarts importants sont observés dans les poids des particules, un indicateur N_{eff} est défini [Liu01], représentant le nombre de particules efficaces

$$N_{\text{eff}} = \frac{1}{\sum_{i=1}^N (w_t^i)^2}$$

Cet indicateur s'interprète comme étant le nombre de particules qui représentent significativement la densité. Idéalement, dans le cas où chacune des particules représente de façon équiprobable la densité, c'est-à-dire avec un poids de $\frac{1}{N}$, alors $N_{\text{eff}} = N$. Cela signifie donc

bien que toutes les particules sont d'intérêt pour l'estimation. Dans le cas extrême où une seule particule représente convenablement la densité et a donc un poids proche de 1, les autres ayant un poids proche de zéro, $N_{\text{eff}} \rightarrow 1$. La définition d'un seuil N_{seuil} à partir duquel le nombre de particules représentatives est considéré comme suffisant permet d'opter pour un rééchantillonnage lorsque l'indicateur N_{eff} lui sera inférieur. La mise en place du schéma complet du filtre particulaire avec échantillonnage d'importance et étape de rééchantillonnage est décrit par l'algorithme 2 et présenté graphiquement par la figure 1.7.

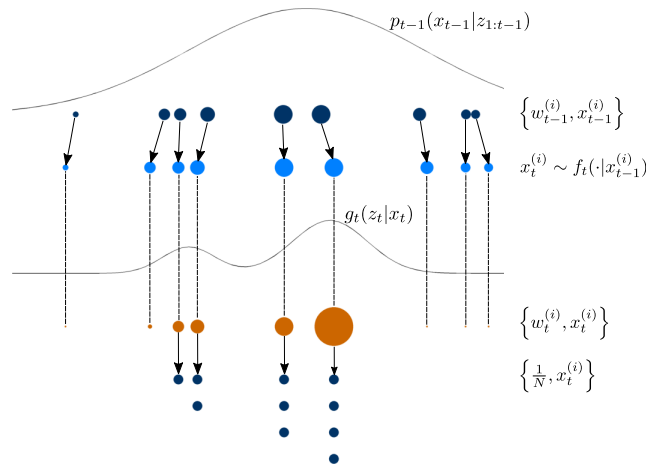


Figure 1.7: Schéma du filtre particulaire au temps t avec rééchantillonnage, et choix de densité de proposition $q(\cdot | x_{t-1}, z_t) = f_t(\cdot | x_{t-1})$

Le choix de la densité de proposition est important car il définit le comportement et donc les performances de l'algorithme. Parmi les choix possibles de densité de proposition, il existe la densité qui minimise la variance des poids des particules tirées, elle est appelée densité optimale (pour ce critère de minimisation). La densité de proposition optimale est la suivante

$$q^{\text{opt}}(x_t | x_{t-1}, z_t) = p(x_t | x_{t-1}, z_t) \quad (1.18)$$

Il est souvent difficile d'échantillonner selon cette densité optimale et d'en avoir une expression explicite, cela n'est possible que dans certains cas particuliers (avec un système linéaire gaussien par exemple). Si cette densité optimale est accessible, en remarquant l'égalité suivante

Algorithme 2 : Filtre particulaire (avec rééchantillonnage adaptatif)

Input : $\mu_0(x_0)$ et $z_{0:t}$
Output : $\{w_t^i, x_t^i\}_{i=1}^N$
 $A \ t = 0$
Initialisation
for $i = 1$ *to* N **do**
 Echantillonner $x_0^i \sim q_0(\cdot)$
 Calculer le poids non normalisé $w_0^i = \frac{\mu_0(x_0^i)}{q_0(x_0^i)}$
end
for $i = 1$ *to* N **do**
 Normaliser le poids $w_0^i = \frac{w_0^i}{\sum_{i=1}^N w_0^i}$
end
while $t \leq T$ **do**
 1. Prédiction
 for $i = 1$ *to* N **do**
 Echantillonner $x_t^i \sim q(\cdot | x_{t-1}^i, z_t)$
 end
 2. Mise à jour
 for $i = 1$ *to* N **do**
 Calculer le poids non normalisé $\tilde{w}_t^i = w_{t-1}^i \frac{g(z_t | x_t^i) f(x_t^i | x_{t-1}^i)}{q(x_t^i | x_{t-1}^i, z_t)}$
 end
 3. Normalisation
 for $i = 1$ *to* N **do**
 Normaliser le poids $w_t^i = \frac{\tilde{w}_t^i}{\sum_{i=1}^N \tilde{w}_t^i}$
 end
 4. Dégénérescence des poids
 Calculer $N_{\text{eff}} = \frac{1}{\sum_{i=1}^N (w_t^i)^2}$
 5. Rééchantillonnage
 if $N_{\text{eff}} \leq N_{\text{seuil}}$ **then**
 Rééchantillonner à partir de $\{w_t^i, x_t^i\}_{i=1}^N$
 end
end

$$\begin{aligned}
g(z_t|x_t)f(x_t|x_{t-1}) &= p(z_t|x_t, x_{t-1})p(x_t|x_{t-1}) \\
&= p(z_t, x_t|x_{t-1}) \\
&= p(z_t|x_{t-1})p(x_t|x_{t-1}, z_t)
\end{aligned} \tag{1.19}$$

alors la pondération des échantillons tirés selon la densité d'importance optimale (1.18) donne, d'après (1.17) et (1.19),

$$w_t^i \propto w_{t-1}^i p(z_t|x_{t-1}^i) \quad \forall i = 1, \dots, N_p \tag{1.20}$$

D'après cette expression (1.20), le poids de la particule i à l'instant courant t ne dépend pas de l'état échantillonné x_t^i de cette particule i tirée d'après la distribution dite optimale. Le poids dépend de l'état et du poids de la particule au temps précédent $t - 1$ et de l'observation qui vient d'être disponible z_t . Ne serait-il donc pas possible de tirer profit de ces informations avant de tirer le nouvel état de la particule ? C'est ce que propose la méthode du filtre particulaire auxiliaire.

1.4.2 Filtre particulaire auxiliaire

Le filtre particulaire auxiliaire, introduit par [PS99], s'appuie sur une variable auxiliaire afin de réaliser l'échantillonnage des particules séquentiellement. La variable auxiliaire vient augmenter la variable d'état, les particules sont échantillonnées selon une loi (jointe) sur la variable augmentée, formant alors des particules du couple variable d'état-variable auxiliaire. Dans la version originalement présentée [PS99], la variable auxiliaire représente l'indice de la particule au temps précédent (dénommée particule ancêtre) à partir de laquelle la variable d'état est propagée. Le tirage de la variable auxiliaire sélectionne une particule ancêtre et introduit donc un rééchantillonnage des particules. L'état de la particule peut ainsi être échantillonné (propagée) à partir de l'état de la particule ancêtre. Un second rééchantillonnage classique des particules après pondération est proposé dans la version initiale de filtre particulaire auxiliaire [PS99]. Ce deuxième rééchantillonnage a été cependant jugé superflu dans des études postérieures [DMO09, JD08].

Les variables auxiliaires viennent augmenter les variables d'état et ainsi faciliter l'échantillonnage à partir de la distribution souhaitée. Les variables auxiliaires peuvent avoir une interprétation physique dans le problème, mais pas nécessairement. L'algorithme d'échantillonnage

par acceptation-rejet est un exemple d'échantillonnage à l'aide d'une variable auxiliaire, puisque l'échantillonnage s'appuie sur une variable uniforme permettant l'acceptation ou le rejet d'un échantillon. Une fois les particules échantillonnées, les échantillons pondérés des variables d'état sont conservés pour l'étape suivante du filtrage séquentiel, tandis que les échantillons des variables auxiliaires sont laissés de côté.

Lorsque la variable auxiliaire correspond à l'indice des particules qui vont être sélectionnées et à partir desquelles les nouvelles particules sont propagées, comme dans la version originale du filtre particulaire auxiliaire, cette sélection est une forme de rééchantillonnage. Sauf indication contraire, ce sera ce cas qui sera traité dans la suite. Le filtre particulaire auxiliaire est donc intimement lié à cette étape de rééchantillonnage.

Le filtre particulaire dit *bootstrap* peut être vu comme un cas particulier d'un filtre particulaire auxiliaire [DMO09]. En effet le rééchantillonnage des particules est effectué après la pondération des échantillons, introduisant ainsi une sélection des indices des particules qui seront reconduites dans la récursion. A l'inverse, le filtre particulaire auxiliaire peut être vu comme un filtre particulaire classique avec rééchantillonnage [JD08], le rééchantillonnage étant effectué sur les particules ancêtres puisque les nouvelles particules ne sont tirées qu'après. Seulement au moment du rééchantillonnage la distribution cible n'est plus la distribution de filtrage, mais une densité cible intermédiaire [CCF99], correspondant à la densité au temps précédent perturbée par une quantité améliorant le rééchantillonnage en l'orientant. Cette orientation permet le choix des particules prometteuses compte tenu de la nouvelle observation rendue disponible à l'instant courant.

Ainsi à l'instant précédent, la distribution est représentée par un ensemble de N_p particules pondérées, avant possible rééchantillonnage, comme suit:

$$p(x_{t-1}|z_{0:t-1}) \approx p_{N_p}(x_{t-1}|z_{0:t-1}) = \sum_{i=1}^{N_p} w_{t-1}^i \delta_{x_{t-1}^i}(x) \quad (1.21)$$

Afin de lutter contre la dégénérescence des poids, c'est donc à ce niveau qu'est proposé un rééchantillonnage des particules, adaptatif ou non afin de lutter également contre l'appauvrissement des échantillons. Encore une fois, on peut remarquer que cette étape de sélection des particules peut être vue comme un échantillonnage à l'aide de la variable auxiliaire représentant les indices des particules. Après rééchantillonnage de M_p particules les poids sont donc tous égaux à $1/M_p$ et M_p peut être égal ou non à N_p . Dans la suite il sera supposé que $M_p = N_p$.

Introduction de la variable auxiliaire

L'approximation (1.21) peut alors être injectée dans l'étape de mise à jour des équations de filtrage afin de former, à l'aide de la formule de Bayes, une approximation de la distribution de filtrage au temps courant t avec l'arrivée de la nouvelle observation z_t , ce qui donne

$$p(x_t|z_{0:t}) \approx p_{N_p}(x_t|z_{0:t}) \propto p(z_t|x_t) \sum_{i=1}^{N_p} w_{t-1}^i p(x_t|x_{t-1}^i) \quad (1.22)$$

L'objectif est donc d'échantillonner des particules selon cette distribution afin de poursuivre l'estimation séquentielle. Il est évidemment difficile voire impossible d'échantillonner selon cette distribution, sauf dans certains cas particuliers. C'est pour cette raison que le filtre particulaire fait appel à une distribution d'importance.

Dans le cas du filtre particulaire auxiliaire, la variable auxiliaire a vient augmenter la variable d'état x_t afin d'aider à l'échantillonnage, en permettant la construction d'une distribution de proposition efficace. La variable augmentée est alors (x_t, a) . Ainsi la distribution objectif de la variable augmentée est définie sur $\mathcal{X} \times \{1, \dots, N_p\}$, à partir de (1.22) qui en est une loi marginale,

$$p_{N_p}(x_t, a|z_{0:t}) \propto p(z_t|x_t) w_{t-1}^a p(x_t|x_{t-1}^a)$$

De même que pour sa marginale (1.22), il n'est pas souvent possible d'échantillonner directement selon cette distribution, une distribution d'importance jointe $q(x_t, a)$ doit donc être déterminée. Quitte à utiliser une variable auxiliaire pour faciliter la tâche, la densité de distribution est choisie afin

- de permettre un rééchantillonnage pertinent des particules,
- en utilisant au mieux l'information disponible au temps courant.

Une fois les particules (pondérées) obtenues $\{w_t^i, x_t^i, a^i\}_{i=1}^{N_p}$ alors les approximations suivantes sont disponibles

$$p_{N_p}(x_t, a|z_{0:t}) = \sum_{i=1}^{N_p} w_t^i \delta_{x_t^i, a^i}(x_t, a) \quad \text{et sa marginale} \quad p_{N_p}(x_t|z_{0:t}) = \sum_{i=1}^{N_p} w_t^i \delta_{x_t^i}(x_t)$$

Echantillonnage d'importance du filtrage particulaire auxiliaire

Afin d'échantillonner la variable auxiliaire (indice ancêtre) et l'état de la particule, la distribution d'importance jointe sur l'espace d'état et l'espace des indices, s'écrit alors

$$q(x_t, a|x_{t-1}, z_t) = \underbrace{\lambda_t^a}_{\text{pondération du mélange}} \underbrace{q(x_t|x_{t-1}^a, z_t)}_{\text{propagation}} \quad (1.23)$$

La distribution d'importance ainsi définie, les particules sont tirées sous la forme de couple indices - états $\{a^i, x_t^i\}_{i=1}^{N_p}$. La pondération est alors la suivante

$$\begin{aligned} w_t^i &\propto \frac{p_{N_p}(x_t^i, a^i|z_{0:t})}{q(x_t^i, a^i|x_{t-1}^{a^i}, z_t)} \quad \forall i = 1, \dots, N_p \\ &= p(z_t|x_t^i) \frac{w_{t-1}^{a^i}}{\lambda_t^{a^i}} \frac{p(x_t^i|x_{t-1}^{a^i})}{q(x_t^i|x_{t-1}^{a^i}, z_t)} \end{aligned} \quad (1.24)$$

L'algorithme 3 présente le pseudo-code du filtre particulaire auxiliaire.

Les choix explicites, dans la formule (1.23), des poids λ_t^a du mélange permettant la sélection des particules ancêtres, et de la distribution de propagation $q(x_t|x_{t-1}^a, z_t)$, sont libres et dépendent de la stratégie adoptée.

Un choix classique pour le filtrage particulaire auxiliaire

Dans la version dite classique du filtre particulaire auxiliaire [PS99], la distribution d'importance est mise sous la forme suivante

$$q(x_t, a|x_{t-1}, z_t) = \underbrace{q(a|x_{t-1}, z_t)}_{\lambda_t^a} q(x_t|x_{t-1}^a, z_t)$$

qui est similaire à l'expression (1.23) en sous-entendant la dépendance de λ_t^a à l'observation z_t et à l'état au temps d'avant x_{t-1} . C'est ainsi, qu'en s'inspirant de la formulation obtenue avec la distribution d'importance optimale, mais en utilisant les expressions auxquelles on a accès, le filtre particulaire auxiliaire classique fait les choix suivants

$$\begin{aligned} \lambda_t^a &= w_{t-1}^a p(z_t|\mu^a) \\ q(x_t|x_{t-1}^a, z_t) &= p(x_t|x_{t-1}^a) \end{aligned}$$

avec μ^a un paramètre représentatif de l'état, comme par exemple $\mu^a = \mathbb{E}[x_t|x_{t-1}^a]$ ou encore

Algorithme 3 : Filtre particulaire auxiliaire

Input : $p_0(x_0)$ et $z_{0:T}$
Output : $\{w_{0:T}^i, x_{0:T}^i\}_{i=1}^{N_p}$

 A $t = 0$
Initialisation
for $i = 1$ *to* N_p **do**

 | Echantillonner $x_0^i \sim q_0(x_0)$

 | Calculer le poids non normalisé $\tilde{w}_0^i = p(z_0|x_0^i) \frac{p_0(x_0^i)}{q_0(x_0^i)}$
end

 Normaliser les poids $w_0^i = \frac{\tilde{w}_0^i}{\sum_{i=1}^{N_p} \tilde{w}_0^i} \quad \forall i = 1, \dots, N_p$
while $t \leq T$ **do**

| 1. Echantillonnage de la variable auxiliaire (indices des particules ancêtres)

 | **for** $a = 1$ *to* N_p **do**

 | | Déterminer le poids de mélange λ_t^a

 | **end**

 | **for** $i = 1$ *to* N_p **do**

 | | Echantillonner a^i d'après les poids $\{\lambda^a\}_{a=1}^{N_p}$

 | **end**

| 2. Propagation / Echantillonnage des états

 | **for** $i = 1$ *to* N **do**

 | | Echantillonner $x_t^i \sim q(x_t|x_{t-1}^{a^i}, z_t)$

| | 3. Pondération

 | | Calculer le poids non normalisé (1.24) $\tilde{w}_t^i = \frac{w_{t-1}^{a^i}}{\lambda^{a^i}} \frac{p(z_t|x_t^i)p(x_t^i|x_{t-1}^{a^i})}{q(x_t^i|x_{t-1}^{a^i}, z_t)}$

 | **end**

 | Normaliser les poids $w_t^i = \frac{\tilde{w}_t^i}{\sum_{i=1}^{N_p} \tilde{w}_t^i} \quad \forall i = 1, \dots, N_p$

 | $t = t + 1$
end

	poids mélange de proposition $\lambda_t^a = \dots$	(*)	distribution d'importance de propagation $q(x_t x_{t-1}^{a^i}, z_t) = \dots$	pondération $w_t^i = \dots$
Bootstrap PF [GSS93]	w_{t-1}^a	F	$p(x_t x_{t-1}^{a^i})$	$p(z_t x_t^i)$
Optimal importance distribution PF [DGA00]	w_{t-1}^a	F	$p(x_t x_{t-1}^{a^i}, z_t)$	$p(z_t x_{t-1}^{a^i})$
Fully adapted APF [PS99]	$w_{t-1}^a p(z_t x_{t-1}^a)$	V	$p(x_t x_{t-1}^{a^i}, z_t)$	1
Classic APF [PS99]	$w_{t-1}^a p(z_t \mu_t^a)$	V	$p(x_t x_{t-1}^{a^i})$	$\frac{p(z_t x_t^i)}{p(z_t \mu_t^{a^i})}$
Optimal weights APF [DMO09]	$w_{t-1}^a \tau_{\text{opt},q}^a$	V	$q(x_t x_{t-1}^{a^i}, z_t)$	$\frac{p(z_t x_t^i)p(x_t^i x_{t-1}^{a^i})}{\tau_{\text{opt},q}^{a^i} q(x_t^i z_t, x_{t-1}^{a^i})}$

Table 1.1: Récapitulatif de choix de filtre particulaire de type auxiliaire. (*) Dans la deuxième colonne est indiqué par V/F (Vrai/Faux) si le rééchantillonnage utilisé permet ou non la prise en compte de la nouvelle observation.

$\mu^a \sim p(x_t|x_{t-1}^a)$. Avec ces choix d'expression, la pondération devient simplement

$$w_t^i \propto \frac{p(z_t|x_t^i)}{p(z_t|\mu_t^i)}$$

La tableau 1.1 présente différents choix possibles et interprétations du filtre particulaire auxiliaire présents dans la littérature. Dans la dernière ligne du tableau, la variable qui permet de minimiser la variance des poids est définie dans [DMO09] par

$$\tau_{\text{opt},q}^a = \left[\int |p(z_t|x) \frac{p(x|x_{t-1}^a)}{q(x|z_t, x_{t-1}^a)}|^2 q(x|z_t, x_{t-1}^a) dx \right]^{1/2}$$

Le filtre particulaire auxiliaire dans sa version appelée classique est adapté aux cas où le bruit dynamique du modèle est faible. En effet, les particules ancêtres sélectionnées sont celles dont on anticipe qu'elles produiront des particules descendantes pertinentes. Il est donc attendu d'observer à nouveau ce comportement lors de l'échantillonnage des particules descendantes.

Le filtre particulaire auxiliaire est utilisé et apporte ses avantages en tant que solution d'implémentation dans des algorithmes de pistage multi-cible en contexte *track-before-detect*, comme dans les papiers [NCM13] et [SCSC16].

Filtre particulaire auxiliaire multidimensionnel et filtrage multi-cible

Le filtre particulaire auxiliaire tel que décrit ci-dessus est défini pour $x \in \mathcal{X}$ pouvant être multidimensionnel. En revanche, tel que défini classiquement, la variable auxiliaire est unidimensionnelle et prend valeur dans l'ensemble $\{1, \dots, N_p\}$ décrivant les numéros des particules¹.

Si la structure du vecteur d'état peut être partagée de façon pertinente, serait-il possible d'avoir une variable auxiliaire multidimensionnelle, et donc de nouvelles particules issues de plusieurs particules ancêtres ? Dans le contexte multicible, le vecteur d'état multicible $X_t = [x_{t,1}, \dots, x_{t,N}]$ est composé des vecteurs d'état des cibles. En considérant les expressions suivantes

- indépendance dynamique des cibles:

$$p(X_t|X_{t-1}) = \prod_{j=1}^N p(x_{t,j}|x_{t-1,j})$$

- approximation particulaire disponible:

$$p_{N_p}(X_{t-1}|z_{0:t-1}) = \sum_{i=1}^{N_p} w_{t-1}^i \delta_{X_{t-1}^i}(X_{t-1})$$

une approximation de la distribution multicible est

$$p_{N_p}(X_t|z_{0:t}) \propto p(z_t|X_t) \sum_{i=1}^{N_p} w_{t-1}^i \prod_{j=1}^N p(x_{t,j}|x_{t-1,j}^i) \quad (1.25)$$

Si aux deux points ci-dessus est ajoutée l'hypothèse d'indépendance postérieure des cibles, c'est-à-dire que le deuxième point est formulé plutôt comme suit

- hypothèse d'indépendance postérieure des cibles:

$$p_{N_p}(X_{t-1}|z_{0:t-1}) = \prod_{j=1}^N \sum_{a_j=1}^{N_p} w_{t-1}^{a_j} \delta_{x_{t-1,j}^{a_j}}(x_{t-1,j})$$

¹ toujours dans le cas où la variable auxiliaire choisie ici représente des indices de particules.

alors l'approximation particulaire de la distribution multicible cherchée s'écrit plutôt

$$p_{N_p}(X_t|z_{0:t}) \propto p(z_t|X_t) \prod_{j=1}^N \sum_{a_j=1}^{N_p} w_{t-1}^{a_j} p(x_{t,j}|x_{t-1,j}^{a_j}) \quad (1.26)$$

et cette écriture, contrairement à celle de (1.25), autorise une sélection différente de la particule ancêtre pour chacune des cibles.

Ainsi l'hypothèse d'indépendance postérieure fait apparaître le produit sur les cibles de mélanges pondérés monocibles. Sous cette forme, il est donc envisageable que les cibles soient échantillonnées à partir d'une particule ancêtre différente. Dans ce cas, la variable auxiliaire représentant l'indice de la particule ancêtre est alors une variable multidimensionnelle de même taille que le nombre de cibles. Cette variable auxiliaire multidimensionnelle ainsi constituée s'écrit donc

$$a = [a_1, \dots, a_j, \dots, a_N] \in \{1, \dots, N_p\}^N$$

De même que précédemment, la variable auxiliaire multidimensionnelle vient augmenter la variable d'état multicible, et ainsi l'approximation de la loi jointe sur l'état augmenté, dont l'expression (1.26) est une marginale, est définie sur $\mathcal{X} \times \{1, \dots, N_p\}^N$ par

$$p_{N_p}(X_t, a|z_{0:t}) \propto p(z_t|X_t) \prod_{j=1}^N w_{t-1}^{a_j} p(x_{t,j}|x_{t-1,j}^{a_j})$$

qui peut être réécrit, afin de faire apparaître une densité d'importance auxiliaire, comme suit

$$\begin{aligned} p_{N_p}(X_t, a|z_{0:t}) &\propto p(z_t|X_t) \prod_{j=1}^N \frac{w_{t-1}^{a_j}}{\lambda_t^{a_j}} \frac{p(x_{t,j}|x_{t-1,j}^{a_j})}{q(x_{t,j}|x_{t-1,j}^{a_j})} \lambda_t^{a_j} q(x_{t,j}|x_{t-1,j}^{a_j}) \\ &= p(z_t|X_t) \underbrace{\prod_{j=1}^N \frac{w_{t-1}^{a_j}}{\lambda_t^{a_j}} \frac{p(x_{t,j}|x_{t-1,j}^{a_j})}{q(x_{t,j}|x_{t-1,j}^{a_j})}}_{\tilde{w}_t^i} \underbrace{\prod_{j=1}^N \lambda_t^{a_j} q(x_{t,j}|x_{t-1,j}^{a_j})}_{q(X_t, a|X_{t-1}, z_t)} \end{aligned} \quad (1.27)$$

A partir de cette écriture, et sous les hypothèses énoncées plus haut, un filtre particulaire auxiliaire multicible peut être imaginé, dont le pseudo-code est décrit dans l'algorithme 4.

Ces croisements entre particules sont également apparus dans les filtres particuliers faisant

Algorithme 4 : Filtre particulaire auxiliaire multicible (avec hypothèse d'indépendance postérieure)

Input : $p_0(X_0)$ et $z_{0:T}$

Output : $\{w_{0:T}^i, X_{0:T}^i\}_{i=1}^{N_p}$

A $t = 0$

Initialisation

for $i = 1$ *to* N_p **do**

 Echantillonner $X_0^i \sim q_0(X_0)$

 Calculer le poids non normalisé $\tilde{w}_0^i = \frac{p_0(X_0^i)}{q_0(X_0^i)}$

end

Normaliser les poids $w_0^i = \frac{\tilde{w}_0^i}{\sum_{i=1}^{N_p} \tilde{w}_0^i} \quad \forall i = 1, \dots, N_p$

while $t \leq T$ **do**

 1. Echantillonnage de la variable auxiliaire (indices des particules ancêtres)

for $j = 1$ *to* N **do**

for $a_j = 1$ *to* N_p **do**

 Déterminer le poids de mélange $\lambda_t^{a_j}$

end

for $i = 1$ *to* N_p **do**

 Echantillonner a_j^i d'après les poids $\{\lambda^{a_j}\}_{a_j=1}^{N_p}$

end

end

 2. Propagation / Echantillonnage des états

for $i = 1$ *to* N_p **do**

for $j = 1$ *to* N **do**

 Echantillonner $x_{t,j}^i \sim q(x_{t,j} | x_{t-1,j}^{a_j^i}, z_t)$

end

3. Pondération

 Calculer le poids non normalisé $\tilde{w}_t^i$ (1.27)

end

Normaliser les poids $w_t^i = \frac{\tilde{w}_t^i}{\sum_{i=1}^{N_p} \tilde{w}_t^i} \quad \forall i = 1, \dots, N_p$

$t = t + 1$

end

appel à des méthodes MCMC. En effet les méthodes SMCMC échantillonnent le nouvel ensemble de particules grâce à une chaîne de Markov, chaque particule étant échantillonnée successivement conditionnée par la précédente. Si l'état multidimensionnel de l'état peut être découpé en blocs pertinents, par exemple un bloc par cible pour un état multicible, alors le noyau de transition de la chaîne de Markov peut proposer une nouvelle particule dont un des blocs est issu d'une particule ancêtre tirée aléatoirement, les autres blocs restant fixés [KBD05, BDB12]. Au fur et à mesure sont constituées des particules dont les blocs proviennent de particules ancêtres différentes. Des méthodes dites évolutives, comme l'opérateur de *crossover* (*crossover operator*) ou l'opérateur d'échange (*exchange operator*)², amènent de la diversité dans les particules ancêtres en échangeant des blocs de particules entre deux chaînes de Markov lancées en parallèle.

Le filtre particulaire auxiliaire suscite donc l'intérêt par sa flexibilité dans le choix des expressions, permettant d'orienter l'échantillonnage en favorisant les particules ancêtres prometteuses. Ainsi la méthode de la variable auxiliaire peut être utilisée pour approcher d'autres distributions que la distribution sur l'état augmenté [EMBD18], et le choix des expressions, notamment des paramètres de pondération du mélange λ_t^a , reste libre.

1.4.3 Filtre particulaire marginalisé (ou *Rao-Blackwellisé*)

Le filtre particulaire marginalisé [SGN05, DdMR00], aussi appelé filtre particulaire rao-blackwellisé, est une méthode s'appliquant aux modèles présentant une partie linéaire et une partie non linéaire. Cette méthode permet d'exploiter la partie linéaire du modèle de manière optimale, tout en utilisant des méthodes particulières pour la partie non linéaire. Le filtrage particulaire est donc appliqué sur un espace de dimension plus petite, en parallèle avec l'estimation linéaire. En résulte une réduction de la variance. Cette méthode est compatible avec les méthodes particulières faisant appel au rééchantillonnage d'importance [GGB⁺02], avec entre autres le filtre particulaire auxiliaire [FSK10], ainsi que celles basées sur les méthodes MCMC [SPGC09].

L'état du système peut être partagé de la manière suivante

$$x_t = \begin{bmatrix} x_{l,t} \\ x_{nl,t} \end{bmatrix} \quad \begin{array}{l} \text{partie linéaire} \\ \text{partie non linéaire} \end{array}$$

tel que, comme dans le cas le plus général présenté dans [SGN05] (modèle 3), le système d'équations dynamiques et d'observation puisse s'écrire de la façon suivante

²ces opérateurs sont issus des méthodes d'algorithmes génétiques

$$x_{1,t} = F_1(x_{n1,t-1}) x_{1,t-1} + f_1(x_{n1,t-1}) + G_1(x_{n1,t-1}) v_{1,t} \quad (1.28a)$$

$$x_{n1,t} = F_{n1}(x_{n1,t-1})x_{1,t-1} + f_{n1}(x_{n1,t-1}) + G_{n1}(x_{n1,t-1})v_{n1,t} \quad (1.28b)$$

$$z_t = H(x_{n1,t-1}) x_{1,t-1} + h(x_{n1,t}) + w_t \quad (1.28c)$$

avec

$$v_t = \begin{bmatrix} v_{1,t} \\ v_{n1,t} \end{bmatrix} \sim \mathcal{N}(0, Q_t), \quad Q_t = \begin{bmatrix} Q_{1,t} & Q_{1/n1,t} \\ Q'_{1/n1,t} & Q_{n1,t} \end{bmatrix}$$

et $w_t \sim \mathcal{N}(0, R_t)$.

L'idée du filtre particulaire marginalisée part de la décomposition suivante

$$p(x_t | z_{0:t}) = p(x_{1,t}, x_{n1,t} | z_{0:t}) = \underbrace{p(x_{1,t} | x_{n1,t}, z_{0:t})}_{\text{filtre de Kalman}} \underbrace{p(x_{n1,t} | z_{0:t})}_{\text{filtre particulaire}}$$

En effet, le conditionnement par la partie non linéaire du vecteur d'état $x_{1,t}$ permet faire apparaître dans le système (1.28) une sous-structure linéaire, avec des bruits additifs Gaussien et permet donc l'utilisation du filtre de Kalman. Quant à la partie non linéaire du vecteur d'état $x_{n1,t}$, un filtre particulaire est appliqué pour estimer la distribution souhaitée. Concrètement, le filtre particulaire permet d'obtenir des échantillons de la partie non linéaire du vecteur d'état. Et pour chacune de ses particules, la partie linéaire du vecteur d'état est donnée selon le filtre optimal de Kalman. Le filtre de Kalman étant déterminé pour la partie linéaire conditionnellement à la partie non linéaire du vecteur d'état, l'échantillonnage des particules de la partie non linéaire est réalisé avant. Il y a un filtre de Kalman pour chaque particule échantillonnée. Le filtre particulaire marginalisé est présenté par l'algorithme 5.

Un cas particulier de système

Dans le cadre de ce manuscrit, l'écriture du système d'intérêt peut se réduire à l'expression ci-dessous, qui est un cas particulier du système général décrit plus haut,

Algorithme 5 : Filtre particulaire marginalisé (ou Rao-Blackwellisé)

Input : $p_0(x_{n1,0}), \{\bar{x}_{1,0}, \bar{P}_0\}$ et $z_{0:T}$
Output : $\{w_{0:T}^i, x_{0:T}^i, P_{0:T}^i\}_{i=1}^{N_p}$
 $A \ t = 0$
Initialisation
for $i = 1$ *to* N_p **do**
 Echantillonner $x_{n1,0}^i \sim q_0(x_{n1,0})$
 Initialiser $\{x_{1,0| -1}^i, P_{0| -1}^i\} = \{\bar{x}_{1,0}, \bar{P}_0\}$
 Calculer le poids non normalisé $\tilde{w}_0^i = p(z_0|x_{n1,0}^i) \frac{p_0(x_{n1,0}^i)}{q_0(x_{n1,0}^i)}$
end
Normaliser les poids $w_0^i = \frac{\tilde{w}_0^i}{\sum_{i=1}^{N_p} \tilde{w}_0^i} \quad \forall i = 1, \dots, N_p$
Rééchantillonnage éventuel
for $i = 1$ *to* N_p **do**
 Calculer $x_{1,0}^i$ et P_0^i [SGN05]
end
 $t = t + 1$
while $t \leq T$ **do**
 for $i = 1$ *to* N **do**
 1. Echantillonnage des particules (partie non linéaire)
 Echantillonner $x_{n1,t}^i \sim q(x_{n1,t}|x_{n1,t-1}^i, z_t)$
 2. Filtre de Kalman - prédiction (partie linéaire)
 Calculer $x_{1,t|t-1}^i$ et $P_{t|t-1}^i$ [SGN05]
 3. Pondération des particules
 Calculer le poids non normalisé $\tilde{w}_t^i = w_{t-1}^i \frac{p(z_t|x_{n1,t}^i)p(x_{n1,t}|x_{n1,t-1}^i, z_{0:t-1})}{q(x_{n1,t}^i|x_{n1,t-1}^i, z_{0:t-1})}$
 end
Normaliser les poids $w_t^i = \frac{\tilde{w}_t^i}{\sum_{i=1}^{N_p} \tilde{w}_t^i} \quad \forall i = 1, \dots, N_p$
Rééchantillonnage éventuel
 4. Filtre de Kalman - mise à jour (partie linéaire)
 for $i = 1$ *to* N_p **do**
 Calculer $x_{1,t}^i$ et P_t^i [SGN05]
 end
 $t = t + 1$
end

$$x_{l,t} = F_l x_{l,t-1} + v_{l,t} \quad (1.29a)$$

$$x_{nl,t} = F_{nl}x_{nl,t-1} + f_{nl}(x_{nl,t-1}) + v_{nl,t} \quad (1.29b)$$

$$z_t = h(x_{nl,t}, e_t) \quad (1.29c)$$

avec e_t le bruit d'observation, pouvant être autre qu'un bruit additif et Gaussien ici.

- Partie linéaire du vecteur d'état: vitesses des cibles (en coordonnées cartésiennes).
- Partie non linéaire du vecteur d'état: positions (en coordonnées cartésiennes).

Remarque – Dans le modèle d'observation choisi ici pour le capteur radar, seules les positions des cibles interviennent dans le traitement du signal radar (traitement en distance et en angle). Si un traitement Doppler était ajouté au traitement du signal radar, une dimension pour la vitesse radiale serait ajoutée aux deux dimensions distance et angle de l'observation radar. Ainsi la vitesse interviendrait dans l'équation d'observation, et de manière non linéaire, et de ce fait la décomposition proposée ici ne serait plus possible.

La formulation (1.29) étant un cas particulier du modèle 3 de [SGN05], il suffit d'adapter leur solution développée pour le système général (1.28) au système particulier (1.29). A cela s'ajoutent deux particularités du modèle (1.29):

- **Modèle du bruit d'observation libre**

La partie linéaire de l'état n'intervient pas dans l'équation d'observation, c'est-à-dire que $H(x_{nl,t-1}) = 0$ dans (1.28c). Cela signifie que la traditionnelle étape de mise à jour par l'observation du filtre de Kalman ne peut être menée (étape 4 de l'algorithme 5). L'équation d'observation n'intervient que dans la partie du filtre particulière. Deux remarques peuvent être alors faites:

- l'hypothèse d'un bruit d'observation additif et gaussien peut être assouplie; le modèle du bruit d'observation est libre, comme écrit dans (1.29c).
- l'information de l'observation n'intervient que dans le filtre particulière pour échantillonner l'état non linéaire. Grâce à l'expression (1.29b), l'état non linéaire devient une sorte d'observation pour l'état linéaire: une étape de mise à jour de l'état linéaire est possible et l'information de l'observation entre le filtre de Kalman via les particules de l'état non linéaire.

- **Matrice de covariance de Kalman unique**

Si l'état linéaire n'apparaît pas dans l'équation d'observation, et si les matrices F_l , F_{nl} ,

G_1 et G_{n1} ne dépendent pas de $x_{n1,t-1}$ alors la matrice de covariance du filtre de Kalman est identique pour chaque particule, c'est-à-dire $\forall i = 1, \dots, N_p, \quad P_t^i = P_t$.

1.5 Méthodes de Monte Carlo par chaîne de Markov

Les méthodes de Monte Carlo par chaîne de Markov ont pour but d'estimer une densité d'intérêt, appelée **densité cible**. Elles s'appuient donc sur la représentation de Monte Carlo donnée dans la partie 1.2, c'est-à-dire une approximation des densités de probabilité recherchées par un ensemble de particules pondérées, en exploitant certaines propriétés des chaînes de Markov, afin d'obtenir des particules représentant la densité cible. Contrairement à l'échantillonnage d'importance où toutes les particules sont indépendamment tirées selon une densité d'importance puis pondérées, les méthodes MCMC vont échantillonner les particules les unes à la suite des autres suivant une chaîne de Markov bien choisie. Plus précisément, la chaîne de Markov va progressivement construire une image de la distribution cible en explorant localement l'espace d'état jusqu'à ce que toutes les régions d'intérêt aient été couvertes. L'indépendance entre les particules ainsi tirées n'est donc pas établie, mais la théorie des chaînes de Markov permet d'établir la convergence de la chaîne vers la distribution cible qui est la loi stationnaire de la chaîne, ainsi que la stabilisation de la moyenne empirique. Le but est donc d'établir la chaîne de Markov dont la loi stationnaire est la loi à approximer, typiquement la densité de filtrage à l'instant t de la figure 1.2. Les deux algorithmes classiques des méthodes MCMC [RC04] sont l'échantillonneur de Metropolis-Hastings [Rob15, CG95], ainsi qu'un de ses cas particuliers, l'échantillonneur de Gibbs [CG92].

Les deux prochaines sections sont tirées de la seconde édition du livre de Christan P. Robert et George Castella [RC04], des chapitres 6 et 7 respectivement.

Rappels sur les chaînes de Markov

Le but de cette section est de donner quelques rappels sur les chaînes de Markov permettant d'établir la convergence de la chaîne vers la densité cible, ce que devront respecter les algorithmes présentés. L'espace d'état est considéré comme étant continu.

Soit $\{X^{(n)}\}_{n \geq 0}$ une suite de variables aléatoires à valeurs dans un ensemble $\mathcal{X}$ (espace d'états). On dit que $\{X^{(n)}\}$ est une **chaîne de Markov** si, conditionnellement aux états passés $X^{(0:n-1)}$, l'état actuel $X^{(n)}$ ne dépend que de l'état immédiatement précédent $X^{(n-1)}$

$$\Pr \left(X^{(n)} \in dx | X^{(0:n-1)} \right) = \Pr \left(X^{(n)} \in dx | X^{(n-1)} \right)$$

et est complètement caractérisée par:

- un **noyau de transition** K défini sur $\mathcal{X} \times \mathcal{B}(\mathcal{X})$

$$K(x, dx') = \Pr \left(X^{(n)} \in dx' | X^{(n-1)} = x \right)$$

- une **distribution initiale** $\mu(dx) = \Pr \left(X^{(0)} \in dx \right)$

ainsi $\forall x \in \mathcal{X}, \forall A \in \mathcal{B}(\mathcal{X}), K(x, A)$ représente la probabilité de se déplacer de x à un point de l'ensemble A .

La densité de transition désigne la densité (si elle existe) $k(x, x')$ du noyau de transition $K(x, \cdot)$

$$\begin{aligned} \Pr(X^{(n)} \in A | X^{(n-1)} = x) &= K(x, A) \\ &= \int_A K(x, dx') \\ &= \int_A k(x, x') dx' \end{aligned}$$

et ainsi $k(x, x') = p(x'|x)$ représente bien la densité conditionnelle de x' connaissant x . Dans la cas d'un espace discret (fini ou dénombrable), ces probabilités de transition définissent une matrice de transition.

Le noyau de transition de x à A en n itérations défini par

$$K^{(n)}(x, A) = \Pr \left(X^{(n)} \in A | X^{(0)} = x \right)$$

coïncide avec la n -ème itérée du noyau de transition K .

L'**irréductibilité** est une mesure de sensibilité de la chaîne de Markov aux conditions initiales. Elle garantit la convergence des algorithmes MCMC. On dit qu'une chaîne est φ -**irréductible** pour une mesure φ si [RC04, Définition 6.13]

$$\exists n, K^{(n)}(x, A) > 0$$

- $\forall x \in \mathcal{X}$
- $\forall A \in \mathcal{B}(\mathcal{X})$ avec $\varphi(A) > 0$

c'est-à-dire que chaque ensemble A a une chance d'être visité par la chaîne de Markov. Cette propriété est trop faible pour assurer que la trajectoire de $\{X^{(n)}\}$ atteindra A assez souvent.

Une chaîne de Markov doit posséder de bonnes propriétés de stabilité pour garantir une approximation acceptable du modèle. La formalisation de cette stabilité mène à différentes notions de récurrence selon que l'espace d'état est discret ou non (ainsi, si l'espace d'état est discret, la récurrence d'un état correspond simplement à une probabilité égale à 1 d'y revenir). Si la chaîne est irréductible, la récurrence devient une propriété de la chaîne. Cette propriété est toujours satisfaite pour les chaînes irréductibles sur des espaces finis. Pour les chaînes à espace d'état continu, la récurrence au sens de Harris (ou Harris récurrence) est définie sur un ensemble, et peut être étendue à une propriété sur la chaîne.

Un ensemble A est Harris récurrent si en simulant la chaîne de Markov à l'infini, on visite A un nombre infini de fois avec une probabilité égale à 1. Le nombre de visite reçues par un ensemble A étant défini par $\eta_A = \sum_{n=1}^{\infty} \mathbb{1}_A(X^{(n)})$, la Harris récurrence se traduit par $\Pr(\eta_A = \infty) = 1$, quelque soit l'état x de départ [RC04, Définition 6.32].

Une chaîne est dite récurrente au sens de Harris (ou **Harris récurrente**) si [RC04, Définition 6.32]

- elle est φ -irréductible
- $\forall A \in \mathcal{B}(\mathcal{X})$ avec $\varphi(A) > 0$, A est un ensemble récurrent au sens de Harris (ou Harris récurrent)

Une chaîne est dite **apériodique** si elle n'est pas **périodique**. Une chaîne φ -irréductible est dite périodique s'il existe une partition de l'espace d'état $\{A_1, \dots, A_d\}$ avec $d \geq 2$, c'est-à-dire $\forall i \neq j : A_i \cap A_j = \emptyset$ et $\cup_{i=1}^d A_i = \mathcal{X}$ telle que

$$\forall i, j, n, m : \quad \Pr \left(X^{(n+m)} \in A_j | X^{(n)} \in A_i \right) = \begin{cases} 1 & j = i + m \bmod m \\ 0 & \text{sinon} \end{cases}$$

c'est-à-dire par exemple si $\{A_1, A_2\}$ est une partition de l'espace d'état $\mathcal{X}$ et que l'ensemble A_1 n'est atteignable par la chaîne de Markov qu'aux itérations impaires, et A_2 aux itérations paires, un phénomène de périodicité 2 est créé.

La densité de transition $k(x, x')$ est dite **réversible** par rapport à la densité de probabilité

p lorsque [RC04, Définition 6.45]:

$$p(x)k(x, x') = p(x')k(x', x) \quad (1.30)$$

Cette équation (1.30) est aussi appelée la condition d'équilibre détaillée.

La densité de transition $k(x, x')$ laisse **invariant** la densité de probabilité p (ou p est **invariant** par k) lorsque:

$$p(x') = \int_{\mathcal{X}} p(x)k(x, x')dx$$

La réversibilité implique l'invariance. Si la densité de transition $k(x, x')$ est réversible par rapport à la densité de probabilité p , alors p est invariant par k . Ce résultat s'obtient en intégrant directement (1.30) sur $\mathcal{X}$ par rapport à x .

La densité de probabilité invariante est unique. Non seulement elle est importante car elle définit un processus stationnaire, mais aussi elle est la mesure de probabilité qui définit le comportement à long terme, ou ergodique, de la chaîne. Ainsi si une distribution limite de la chaîne existe, il s'agit d'une mesure de probabilité invariante, et comme cette dernière est unique, elle est également indépendante de l'état initial de la chaîne.

Si la chaîne de Markov associée à la densité k est:

- Harris récurrente (donc nécessairement p -irréductible)
- apériodique
- p -invariante

alors on a la convergence suivante [RC04, Théorème 6.51] $\forall x \in \mathcal{X}$

$$\Pr \left(X^{(n)} \in A | X^{(0)} = x \right) \xrightarrow{n \rightarrow \infty} p(A)$$

avec par définition $p(A) = \int_A p(x)dx$.

Echantillonneur de Metropolis-Hastings

L'algorithme de Metropolis-Hastings (MH) s'appuie sur les propriétés suffisantes citées ci-dessus afin de construire une chaîne de Markov, à l'aide d'une densité de transition $k(x, x')$ bien élaborée, dont la distribution invariante p , celle vers laquelle la chaîne converge, est la

distribution cible à estimer. En simulant la chaîne suffisamment longtemps pour que son comportement soit stationnaire, les échantillons issus de la chaîne approchent la densité invariante. Comme pour la méthode d'échantillonnage d'importance, on ne sait bien sûr pas échantillonner selon p , mais on connaît son expression à une constante multiplicative près.

La densité de Metropolis-Hastings $k(x, x')$ doit remplir les conditions permettant la convergence de la chaîne de Markov vers la distribution invariante. Il s'appuie sur une densité connue, $q(x, x')$, telle que:

1. il est facile d'échantillonner selon $q(x, \cdot)$ pour tout x
2. $\text{support}(p) \subset \text{support}(q)$
3. l'expression de $q(x, x')$ est connue (à une constante multiplicative près ne dépendant pas de x)

La densité q va être "perturbée", par une densité d'acceptation, afin de constituer une densité $k(x, x')$ qui soit réversible par rapport à la densité cible p (et donc invariante par cette même densité), permettant ainsi d'assurer la convergence de la chaîne de Markov vers p .

La densité markovienne k de Metropolis-Hastings est définie par

$$k(x, x') = \alpha(x, x')q(x, x') + (1 - \alpha(x))\delta_x(x') \quad (1.31)$$

avec

$$\alpha(x, x') = \min \left(1, \frac{p(x')q(x', x)}{p(x)q(x, x')} \right) \quad \text{et} \quad \alpha(x) = \int_{\mathcal{X}} \alpha(x, y)q(x, y)dy \quad (1.32)$$

- la transition donnée par $q(x, x')$ est acceptée avec la probabilité $\alpha(x, x')$
- la transition reste en x avec la probabilité $1 - \alpha(x)$

Alors la condition d'équilibre détaillée (1.30) pour la densité de Metropolis-Hastings (1.31) s'écrit

$$p(x)q(x, x')\alpha(x, x') = p(x')q(x', x)\alpha(x', x)$$

et est directement vérifiée avec l'expression de $\alpha(x, x')$ (1.32) puisque

$$p(x)q(x, x')\alpha(x, x') = \min(p(x)q(x, x'), p(x')q(x', x)) = p(x')q(x', x)\alpha(x', x)$$

et

$$p(x)(1 - \alpha(x))\delta_x(x') = p(x')(1 - \alpha(x'))\delta_{x'}(x)$$

Ainsi la densité de transition $k(x, x')$ est réversible par rapport à la densité cible $p(x)$, donc la densité cible $p(x)$ est invariante pour l'algorithme de Metropolis-Hastings.

Pour simuler une variable X' selon $k(x, \cdot)$, où x est fixé, il suffit

- de simuler une v.a. X^* distribuée selon $q(x, \cdot)$: $x^* \sim q(x, \cdot)$
- et de poser

$$x' = \begin{cases} x^* & \text{avec probabilité } \alpha(x, x') \\ x & \text{avec probabilité } (1 - \alpha(x, x')) \end{cases}$$

Algorithme 6 : Echantillonneur de Metropolis-Hastings

Input : B burn-in et N nombre d'échantillons

Output : $\{x^i\}_{i=1}^N$

Initialisation

Echantillonner $x^0 \sim \mu(\cdot)$

Itération

for $n = 1$ *to* $B + N$ **do**

 Sachant x^{n-1} , générer un candidat $x^* \sim q(x^{n-1}, \cdot)$

 Calculer la probabilité d'acceptation $\alpha = \alpha(x^{n-1}, x^*)$

 Poser

$$x^{(n)} = \begin{cases} x^* & \text{avec probabilité } \alpha \\ x^{(n-1)} & \text{avec probabilité } (1 - \alpha) \end{cases}$$

end

Garder les N derniers échantillons pour former $\{x^i\}_{i=1}^N$

La période dite de burn-in sert à écarter les premiers échantillons qui sont fortement influencés par l'initialisation de l'algorithme. Lorsque la chaîne devient ensuite stationnaire, on extrait des échantillons de la chaîne. Afin de réduire la dépendance entre les échantillons due à la formation de la chaîne de Markov, seul un nombre choisi d'échantillons peuvent être conservés, par exemple un sur deux. Le choix du noyau de proposition est déterminant. Il doit permettre une exploration suffisante de l'espace d'état afin d'emmener la chaîne vers toutes les régions à forte densité de probabilité.

Echantillonneur de Gibbs

L'échantillonneur de Gibbs est un cas particulier de l'algorithme de Metropolis-Hastings dans le cas d'un vecteur d'état multi-dimensionnel, et dans lequel la densité de proposition $q(x, x')$

est la loi marginale conditionnelle de la loi jointe sur chacune des composantes. Cette stratégie permet d'accepter tous les échantillons générés, la probabilité d'acceptation de Metropolis-Hastings se réduisant à 1. On suppose que x est composé de d composantes: $x = (x_1, \dots, x_d)^T$. A chaque étape n de l'algorithme de Gibbs, chaque composante est échantillonnée à son tour conditionnellement aux valeurs des autres, sous forme de cycle systématique:

$$\begin{aligned} x_1^n &\sim p(x_1 | x_{2:d}^{n-1}) \\ x_2^n &\sim p(x_2 | x_1^n, x_{3:d}^{n-1}) \\ &\vdots \\ x_j^n &\sim p(x_j | x_{1:j-1}^n, x_{j+1:d}^{n-1}) \\ &\vdots \\ x_d^n &\sim p(x_d | x_{1:d-1}^n) \end{aligned}$$

Chaque nouvelle valeur pour chaque composante est immédiatement utilisée dès qu'elle est échantillonnée. Un tirage aléatoire de l'ordre dans lequel les composantes sont traitées est une alternative. L'échantillonneur de Gibbs possède les avantages d'accepter toutes les propositions, et de ne pas nécessiter la définition d'un noyau de transition. Cependant, il faut être capable de savoir échantillonner selon la distribution conditionnelle. Dans certains cas, les composantes sont trop dépendantes pour que l'échantillonneur de Gibbs soit efficace. Il est parfois envisageable d'échantillonner selon une distribution de proposition conditionnelle sur des blocs de composantes plutôt que sur chaque composante individuelle.

Chapitre 2

Pistage multicible et Track-before-detect

Les problématiques du pistage de cibles sont multiples. Le filtrage est le terme désignant les méthodes d'extraction des informations utiles parmi les mesures reçues via le ou les capteurs mis en place pour observer les cibles. De ces informations, le pistage s'attache à détecter les cibles (détection), leur donner une trajectoire (estimation) à chacune (étiquetage). Donner l'estimation d'une trajectoire induit un étiquetage puisque chaque état de la trajectoire doit appartenir à la même entité. Le système composé par les cibles est un système dynamique, c'est-à-dire que l'état des cibles évolue dans le temps. Le filtrage utilise les mesures, ou observations, fournies pour retrouver l'état du système. Ces mesures présentent des défauts, liés à l'environnement d'observation et au fonctionnement interne des capteurs, qu'il est nécessaire de corriger. Ces défauts sont donc des signaux indésirables et parasites, issus du paysage environnant et d'autres objets que l'étude ne concerne pas, ou des perturbations aléatoires, liées aux aspects thermodynamiques des circuits composants les capteurs. Le filtre s'appuie également sur une connaissance a priori du modèle dynamique du système: ce modèle n'est pas déterministe et comporte lui aussi des incertitudes.

Il existe plusieurs types d'observations qui mènent à des problématiques différentes lors de la définition explicite des équations de filtrage. Sont différenciées l'approche traditionnelle du pistage par des **observations ponctuelles**, parfois référée comme la **poursuite pendant le balayage** (track while scan, TWS) et l'approche sur **observations brutes** dite **pistage avant détection** (track before detect, TBD).

- Les observations ponctuelles sont des points donnés dans l'espace d'observation. Cet ensemble de points est issu d'une première détection au niveau du traitement du signal capteur qui nécessite un pré-traitement. Par exemple, dans une image caméra, il va s'agir du milieu du rectangle dans lequel l'objet cible a été trouvé, dans une image radar ce sera un emplacement où le signal réfléchi est élevé et dépasse un seuil, pouvant être interprété comme la présence d'un objet réflecteur à cet endroit. L'observation est donc un ensemble de points, dont certains appartiennent à une cible présente. D'autres points ne correspondent pas à une cible à pister, mais sont dus à l'environnement, au paysage, à des défauts du capteur; ces points sont les **fausses alarmes**. Il se peut également que des cibles n'aient pas généré de points, il s'agit des **détections manquées**. Ces notions permettent de définir une **probabilité de fausse alarme** (probabilité d'observer une fausse alarme) et une **probabilité de détection** (probabilité de détecter la cible, à partir de laquelle on peut retrouver la probabilité de ne pas la détecter, correspondant à la détection manquée citée ci-dessus).
- Les observations brutes quant à elles ne nécessitent pas de pré-traitement du signal reçu du capteur. Il n'y a pas d'effet de seuillage comme précédemment, mais en tout point de l'espace d'observation les données du signal sont transmises et peuvent être traitées. Par exemple, dans une image caméra, il va s'agir de tout les pixels et leurs valeurs composant l'image, dans un image radar ce sera une intensité de signal dans les cases distance-azimut du champ de vision du radar.

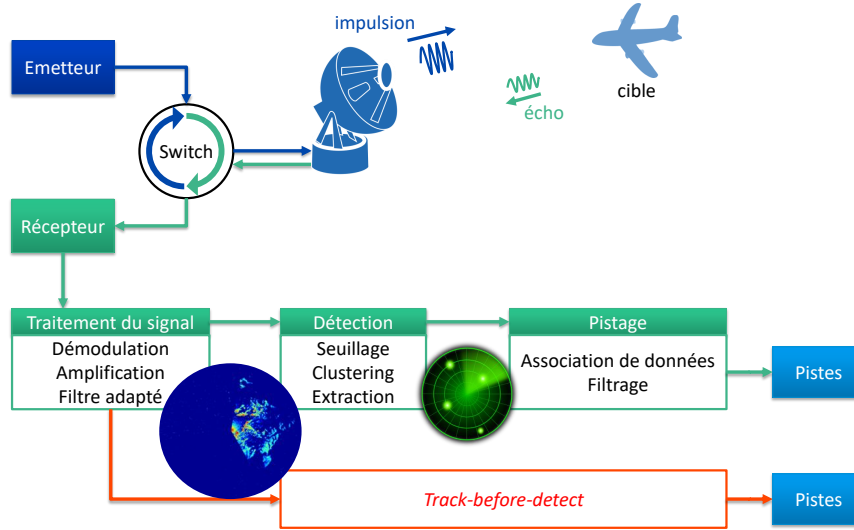


Figure 2.1: Etapes de la poursuite de cibles avec radar

2.1 Observations ponctuelles

2.1.1 Méthodes historiques

Les méthodes dites traditionnelles [BSF88, Bla86] traitent le filtrage monocible et multicible dans le cadre d'observations ponctuelles (seuillées). Elles ont souvent été conçues dans un premier temps pour des systèmes aux équations linéaires et à statistique gaussienne.

Pistage monocible

Le filtrage monocible, ou pistage monocible, s'attache au cas où une seule cible est présente, pouvant être ou non détectée, en présence ou non de fausses alarmes. Le cas le plus simple par exemple est celui où le capteur fournit à chaque étape une seule observation ponctuelle bruitée correspondant à la cible présente qui est alors détectée et dont il reste à estimer l'état. Le scénario se complique lorsque déjà la question de la présence d'une cible se pose, ou bien que la cible ne fournit pas systématiquement une observation à chaque étape (détection manquée), et qu'un nombre aléatoire de fausses alarmes est possible. Dans ce cas, il n'est plus direct de savoir si la cible est présente ou non, ni de savoir avec quelle mesure doit être mise à jour l'état de la cible. C'est ainsi que sont apparues des stratégies d'**association de données**, pour une fois l'association cible-observation effectuée pouvoir utiliser les équations classiques, comme un

filtre de Kalman par exemple lorsque le modèle du système est linéaire gaussien. De ce point de vue, le filtrage monocible peut être vu comme un cas particulier du filtrage multicible. Une stratégie d'association de données basique est celle du **plus proche voisin** [BSF88], "plus proche" étant défini selon un critère de distance (distance euclidienne, distance de Mahalanobis). L'observation la plus proche de l'état prédit de la cible lui est attribuée pour l'étape de mise à jour. Cette méthode simpliste est très vite limitée et donne des résultats médiocres dans le cas où l'observation issue de la cible est noyée dans de nombreuses fausses alarmes, ou bien lors d'une détection manquée. Le filtre par **association de données probabiliste** (probabilistic data association, PDA) [BSF88] quant à lui, n'utilise pas que l'observation la plus proche, mais s'appuie sur toutes les observations disponibles en les fusionnant pour obtenir une seule observation. C'est cette observation qui est utilisée ensuite pour l'étape de mise à jour de l'état de la cible. Une probabilité d'association est calculée pour chaque observation, il s'agit de la probabilité que l'observation ait été générée par la cible, prenant en compte la vraisemblance d'une observation et la probabilité de détection de la cible. L'observation fusionnée est alors la moyenne de toutes les observations pondérées par leur probabilité d'association. Est ensuite introduite la problématique de l'existence de la cible dans la scène, permettant d'établir l'initiation et la terminaison d'une trajectoire, à l'aide d'une approche de modèle multiple en interaction [BBS88] combinée au PDA [HBS89], ou en intégrant la probabilité d'existence directement dans l'algorithme d'association (integrated PDA, IPDA) [MES94].

Pistage multicible

Cette fois, dans le pistage multicible (multi-target tracking, MTT), plusieurs cibles peuvent coexister dans la même scène. Leur nombre peut être constant et connu, dans ce cas seule l'estimation de l'état des cibles reste à définir. Ou bien il peut être constant mais inconnu, et doit être donc estimé conjointement à l'état des cibles. Il peut également être variable, autorisant l'apparition et la disparition des cibles. Clairement, dans le cas multicible, la problématique d'**association de données** apparaît, c'est-à-dire constituer des paires cible-observation correspondantes afin de pouvoir appliquer les techniques de pistage classique. Les observations peuvent être issues chacune d'une des cibles, constituées des fausses alarmes, ou bien les cibles n'ont pas forcément toutes généré des observations. Souvent, l'hypothèse selon laquelle une cible ne peut générer au maximum qu'une observation, est retenue. Il arrive que cette hypothèse soit assouplie, autorisant une cible à générer aucune ou plusieurs observations, pour garantir certaines propriétés, notamment d'indépendance, entre des variables aléatoires mises en jeu dans la stratégie de filtrage considérée. Concernant l'association de données, à nouveau une stratégie dite du **plus proche voisin** peut être proposée. Sous l'hypothèse

qu'une cible ne peut générer au maximum qu'une observation, cette stratégie prend la forme d'un problème d'affectation, c'est-à-dire que l'association entre les mesures et les cibles qui minimise ou maximise la fonction de coût est choisie. La fonction de coût peut être la somme des distances à minimiser, ou bien la somme des probabilités de vraisemblance à maximiser. L'affectation optimale est alors choisie comme solution, et les équations du filtrage monocible peuvent être appliquées à chaque paire cible-observation. Une autre solution propose d'énumérer exhaustivement toutes les associations possibles et de créer ainsi de nombreuses hypothèses, dont le nombre augmente exponentiellement avec le temps, il s'agit du **pistage à hypothèses multiples** (multiple hypothesis tracking, MHT) [Bla86, Rei79]. Pour pallier ce problème d'hypothèses croissantes qui ne peuvent pas être toutes stockées, les hypothèses peu probables sont écartées, et les hypothèses proches combinées, afin de ne garder qu'un nombre raisonnable d'hypothèses à propager temporellement. Les deux stratégies ci-dessus considèrent l'association de données comme déterministe. Deux autres stratégies définissent l'association de données cible-observation comme étant une variable aléatoire. Le filtre par **association de données probabiliste jointe** (joint probabilistic data association, JPDA) [BSF88] est une extension du monocible PDA, dans lequel, l'hypothèse qu'une seule cible est à l'origine d'une observation est faite, créant une dépendance. Une version proposant d'intégrer une probabilité d'existence de chaque cible individuellement (joint integrated probabilistic data association, JIPDA) [ME04] existe également permettant l'initiation et la terminaison des pistes. Contrairement au MHT qui pour chaque hypothèse affecte une association bien définie (une cible pouvant être associée à une observation, ou non en cas de non détection, l'observation pouvant être une fausse alarme), le **pistage à hypothèse multiples probabiliste** (probabilistic multi-hypothesis tracker, PMHT) [SL94] estime la probabilité que chaque observation soit générée par chaque cible, en posant cette probabilité comme variable latente non observable et en résolvant ce problème grâce à l'algorithme espérance-maximisation (expectation-maximization, EM) [DLR77]. Ce filtre s'affranchit de la dépendance citée ci-dessus dans le cas du JAPD. Une comparaison entre ces deux derniers filtres est présentée dans [RWS95] dans le cas où le nombre de cibles est fixé et connu. Une fois l'association de données réalisée, les équations monocibles s'appliquent.

Pour résumé, les méthodes ci-dessus font avant tout appel à des stratégies pour gérer les associations entre mesures et cibles, pour trouver la meilleure association, ou utiliser un mélange pondéré de toutes les associations possibles, en incluant les situations de détections manquées ou de fausses alarmes. Enfin les équations bien connues du filtre de Kalman sont appliquées pour les systèmes aux équations linéaires et à statistique gaussienne, ou ses alternatives (filtre de Kalman étendu et sans parfum) dans le cas où les systèmes ne répondent pas aux contraintes énoncées ci-avant. Cependant, les cas de systèmes non linéaires et à

statistique non gaussienne ne sont pas toujours bien gérés avec les alternatives du filtre de Kalman. Le filtre particulaire octroie lui cette souplesse dans le modèle. Dans le cas multicible avec association de données, des extensions du filtre bootstrap, comme le filtre bootstrap hybride [Gor97], sont proposés. Un algorithme générique de pistage avec association de données est proposé dans [HLCP02b] avec un nombre fixé et connu de cibles, et son extension [HLCP02a] pour un nombre variable de cibles. Il est appliqué dans ces publications au pistage de cibles avec sonar passif. Sous les hypothèses du PMHT, l'association entre les observations et les cibles est réalisée grâce à un échantillonneur de Gibbs.

2.1.2 Cadre RFS

Au delà des ces méthodes traditionnelles est apparue une nouvelle formulation du problème avec les processus ponctuels (point process, PP) [DVJ03, DVJ08] ou les ensemble aléatoires finis (random finite set, RFS), définis dans le cadre de la statistique des ensembles finis (finite set statistics, FISST) développé en détail dans [GMN97]. La seconde formulation, qui peut être considérée comme un cas particulier de la première (le cas du processus ponctuel simple), s'est vu largement développée ces dernières années, notamment dans son application au pistage de cibles, donnant naissance à une panoplie d'algorithmes. Les premiers filtres issus de ce cadre sont le probability hypothesis density filter (PHD) [Mah03] et le cardinalized probability hypothesis density filter (CPHD) [Mah07a]. La formulation RFS considère que les états des cibles et les observations forment chacun un ensemble fini dont la taille est variable, dont les éléments ne sont donc pas ordonnés, et chaque élément est un vecteur d'état d'une cible individuelle (ensemble multicible) ou un vecteur d'une observation (ensemble des observations). Le nombre de cibles, leurs états, le nombre d'observations et leurs valeurs, sont issus du processus stochastique modélisant le système, donc il est nécessaire de définir le concept des ensembles aléatoires finis (RFS). Cette représentation diffère de la représentation classique dans laquelle le vecteur état multicible à estimer est la concaténation des états individuels de chacune des cibles, comme dans le tableau 2.1.

Description de RFS

Ainsi un RFS est une variable aléatoire dont les réalisations sont des ensembles dont la taille et les éléments sont aléatoires. La taille prend valeur dans l'ensemble des entiers naturels correspondant au nombre d'éléments n qui constituent l'ensemble. Chaque élément prend valeur dans l'espace d'état individuel. Il est nécessaire de transposer les notions de théories de la mesure, utilisées pour l'établissement des équations de filtrage bayésien, aux nouveaux

	Cadre classique	Cadre RFS
Etat multicible	$X = [x_1^T, \dots, x_n^T]^T$	$X = \{x_1, \dots, x_n\}$
Observations (ponctuelles)	$Z = \{z_1, \dots, z_m\}$	$Z = \{z_1, \dots, z_m\}$
Association de données	Explicite	Implicite

x_i vecteur état de la cible i , n nombre de cibles
 z_j vecteur de l'observation j , m nombre d'observations

Table 2.1: Notations cadre classique et cadre RFS

objets que sont les RFS [Vo08], permettant ainsi de définir des distributions de probabilités multicibles, et les densités associées si existantes. Les distributions multicibles des RFS sont toujours définies comme des fonctions symétriques, c'est-à-dire que les permutations d'une réalisation donnée se produisent avec la même probabilité, conservant ainsi le caractère non ordonné des éléments du RFS. Par exemple pour un RFS admettant une densité multicible dont une réalisation peut être $X = \{x_1, x_2\}$, alors $p(\{x_1, x_2\}) = p(\{x_2, x_1\})$. Un RFS est donc caractérisé par la distribution (discrète) de sa taille (ou cardinalité), et la distribution spatiale des objets. La définition de ces deux distributions permet de décrire les RFS qui sont utilisés pour déterminer les filtres dans la suite.

• Bernoulli RFS

Le Bernoulli RFS traite le cas monocible où l'on peut avoir une ou zéro cible. La cible existe avec une probabilité r . C'est-à-dire que l'ensemble peut être vide (de cardinalité 0 avec une probabilité $1 - r$), ou ne contenir qu'un seul élément (de cardinalité 1 avec une probabilité de r). La probabilité d'avoir une cardinalité strictement supérieure à 1 est nulle. Si la cible existe, son état est distribué selon une densité spatiale s sur l'espace d'état individuel. Le Bernoulli RFS est donc complètement caractérisé par (r, s) . En notant X la variable représentant le Bernoulli RFS, la densité multicible s'écrit alors

$$p(X) = \begin{cases} 1 - r & \text{si } X = \emptyset \\ r \times s(x) & \text{si } X = \{x\} \\ 0 & \text{si } |X| > 1 \end{cases}$$

• Multi-Bernoulli RFS

Le multi-Bernoulli RFS est l'extension multicible du Bernoulli RFS présenté précédemment. Le multi-Bernoulli RFS est l'union d'un nombre fixe M de Bernoulli RFS notés $X^{(i)}$, indépendants les uns des autres et chacun caractérisé par le couple $(r^{(i)}, s^{(i)})$. En notant X la variable du multi-Bernoulli RFS, on a

$$X = \bigcup_{i=1}^M X^{(i)}$$

Le multi-Bernoulli RFS est complètement caractérisé par $\{(r^{(i)}, s^{(i)})\}_{i=1}^M$. La densité multicible s'écrit alors

$$p(X) = \begin{cases} \prod_{i=1}^M (1 - r^{(i)}) & \text{si } X = \emptyset \\ \prod_{i=1}^M (1 - r^{(i)}) \sum_{1 \leq j_1 \neq \dots \neq j_n \leq M} \prod_{i=1}^M \frac{r^{(j_i)} s^{(j_i)}(x_i)}{1 - r^{(j_i)}} & \text{si } X = \{x_1, \dots, x_n\} \text{ et } n \leq M \\ 0 & \text{si } n > M \end{cases} \quad (2.1)$$

La distribution de la cardinalité d'un multi-Bernoulli RFS est obtenu en écartant la densité spatiale dans (2.1) et donne

$$\rho(n) = \begin{cases} \prod_{i=1}^M (1 - r^{(i)}) & \text{si } n = 0 \\ \prod_{i=1}^M (1 - r^{(i)}) \sum_{1 \leq j_1 \neq \dots \neq j_n \leq M} \prod_{i=1}^M \frac{r^{(j_i)}}{1 - r^{(j_i)}} & \text{si } n \leq M \\ 0 & \text{si } n \geq M \end{cases}$$

- **Independent identically distributed cluster RFS (i.i.d. cluster RFS)**

La distribution de la cardinalité est $\rho(n)$, laissée ici sous sa forme générique. Les éléments du RFS sont tous indépendants et identiquement distribués (i.i.d.) selon la densité spatiale $s(x)$. La densité multicible de l'i.i.d. cluster RFS X s'écrit alors

$$p(X) = \begin{cases} \rho(0) & \text{si } X = \emptyset \\ n! \rho(n) \prod_{x \in X} s(x) & \text{si } X = \{x_1, \dots, x_n\} \end{cases}$$

Le i.i.d cluster RFS apparaît notamment dans la dérivation du filtre CPHD.

- **Poisson RFS**

Il s'agit du cas particulier d'un i.i.d. cluster RFS dans lequel la distribution de la cardinalité est une distribution de Poisson de paramètre λ , c'est-à-dire

$$\rho(n) = \frac{\lambda^n e^{-\lambda}}{n!}$$

La paramètre λ étant la moyenne il s'agit du nombre d'éléments attendus. La densité multicible du Poisson RFS X est alors

$$p(X) = \begin{cases} e^{-\lambda} & \text{si } X = \emptyset \\ \lambda^n e^{-\lambda} \prod_{x \in X} s(x) & \text{si } X = \{x_1, \dots, x_n\} \end{cases}$$

Le Poisson RFS apparaît notamment dans la dérivation du filtre PHD.

Ces méthodes gèrent naturellement le cas multicible, la description de la cardinalité étant complètement intégrée dans la description du RFS. L'association de données n'est pas non plus explicite et déterministe, mais intégrée.

Filtres PHD et CPHD

Les filtres PHD [Mah03] et CPHD [Mah07a] représentent des approximations du premier moment du filtre bayésien multicible. Le filtre PHD estime conjointement le nombre de cibles et leurs états en propageant le moment du premier ordre d'un RFS. Le filtre CPHD propage également le moment du premier ordre d'un RFS, et simultanément la distribution de la cardinalité du RFS. Le filtre CPHD propage donc de l'information supplémentaire mais au prix d'un coût de calcul plus élevé que le PHD. Ce premier moment de la densité multicible s'avère être une fonction d'intensité définie sur l'espace d'état d'une cible individuelle. Cette fonction d'intensité, appelée également PHD et propagée séquentiellement, représente l'espérance mathématique du nombre de cibles. L'intégration de cette fonction d'intensité sur une région de l'espace d'état donne le nombre de cibles attendu dans cette région, l'intégration sur l'espace d'état tout entier donne le nombre total de cibles estimé. Les maxima locaux donnent alors les états estimés des cibles. Pour dériver l'expression de la fonction d'intensité PHD, l'hypothèse est faite que l'ensemble des cibles prédites est un Poisson RFS, et que l'ensemble des fausses alarmes est un Poisson RFS. Le filtre PHD observe une variance élevée pour l'estimation de la cardinalité des cibles. Pour le filtre CPHD, l'hypothèse selon laquelle les cibles prédites et les fausses alarmes sont des Poisson RFS est relaxée, considérant des i.i.d cluster RFS plutôt;

gardant ainsi une expression générique de la distribution de la cardinalité. Le filtre CPHD est donc une généralisation du filtre PHD. Ainsi le filtre CPHD propage conjointement la fonction d'intensité PHD, et la distribution de la cardinalité. On préférera s'appuyer sur la distribution de la cardinalité pour en extraire une estimation de nombre de cibles, plutôt que d'intégrer la fonction d'intensité sur l'espace d'état. L'état des cibles est estimé de même ensuite à partir des maxima locaux de la fonction PHD. Il est possible pour ces deux filtres d'obtenir des expressions explicites des fonctions à propager pour le cas d'un modèle linéaire et gaussien, lorsque les probabilités de détection et de survie des cibles sont des constantes (indépendantes des cibles). Il existe également des implémentations en mélange de gaussiennes (PHD [VM06], CPHD [VVC07]) ou avec un filtre particulaire (PHD [VSD05], CPHD [Vo08]). La version du filtre particulaire, nécessite une étape supplémentaire de clustering afin de donner les états estimés des cibles. Il existe des variantes du PHD dans lesquelles sont introduites d'autres distributions pour la cardinalité des RFS que la distribution de Poisson. Par exemple dans [SDHC16] il est supposé que le nombre de fausses alarmes est distribué selon une distribution binomiale négative, ou bien une distribution de Panjer dans [SDHC18].

Filtres multi-Bernoulli

Des filtres basés sur les multi-Bernoulli RFS ont également été développés, proposant une approximation du filtre bayésien multicible. Il ne s'agit plus comme pour les filtres PHD et CPHD de propager un résumé de l'information statistique que sont le premier moment et la distribution de la cardinalité, mais d'approcher la densité multicible postérieure que l'on propage séquentiellement par des paramètres propagés séquentiellement. Ces paramètres sont ceux décrivant le multi-Bernoulli RFS caractérisant la population des cibles, c'est-à-dire $\left\{ (r_t^{(i)}, s_t^{(i)}) \right\}_{i=1}^{M_t}$, et la récursion est illustrée sur la figure 2.2. Une première version d'un tel filtre est apparu avec le multi-target multi-Bernoulli filter (MeMber) dans [Mah07b], dans lequel l'estimation de la cardinalité est biaisée. Une version non biaisée nommée cardinality balanced MeMber (CBMeMber) est proposée dans [Vo08, VVC09], avec ses implémentations en mélange de gaussiennes et filtre particulaire.

RFS étiquetés (labeled RFS)

Les différentes techniques RFS présentées ci-dessus sont adéquates pour le filtrage multicible et l'estimation de l'état multicible à chaque pas de temps. Cependant, les objets n'ont pas une identité qu'ils gardent dans le temps, permettant de les identifier, en témoignent les expressions des densités multicibles qui sont symétriques, ne donnant aucun ordre aux objets. Or pour réaliser du pistage multicible, c'est-à-dire faire une extraction de pistes (ou trajectoires), il

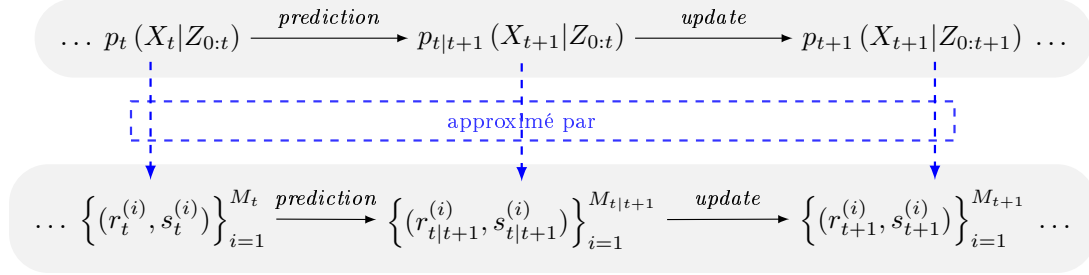


Figure 2.2: Récursions du filtre bayésien et du multi-target multi-Bernoulli filter

est nécessaire de donner une identité aux objets, de les étiqueter, pour savoir à quelle piste appartient l'état estimé d'une cible extrait de l'état multicible. L'idée est alors d'augmenter l'état x d'un objet avec un label (ou étiquette) ℓ unique permettant de l'identifier. On parle alors d'état étiqueté $\mathbf{x} = (x, \ell)$. L'état multicible étiqueté est donc un ensemble d'états étiquetés d'objets. Pour que cet étiquetage soit efficace, il est bien sûr nécessaire que les labels soient uniques pour chacune des cibles composant l'état multicible. L'introduction et la définition des RFS étiquetés (labeled RFS) [VV13] permet de décrire un état multicible étiqueté garantissant des labels distincts pour chaque objet. On note $\mathcal{L}(X)$ l'ensemble des labels du RFS étiqueté X . Les labels d'une réalisation X d'un RFS étiqueté sont distincts si le nombre de labels est égal à la taille (à la cardinalité) de l'état multicible. On peut ainsi définir un indicateur de labels distincts $\Delta(X) = \delta_{|X|}(|\mathcal{L}(X)|)$ vérifiant bien qu'aucune cible ne partage le même label. Cet indicateur est alors utilisé pour déterminer les densités de probabilités multicibles.

Les RFS étiquetés ont largement été développés pour étendre les RFS multi-Bernoulli [Reu14] en développant plusieurs versions, ainsi que les filtres multi-Bernoulli leur correspondant.

2.2 Observations brutes

L'utilisation d'observations ponctuelles requiert peu de mémoire de stockage des mesures et peut être efficace avec un coût de calcul peu élevé. Cependant le pré-traitement de détection afin d'extraire les mesures ponctuelles perd de l'information, notamment dans les cas de cibles à faible rapport signal sur bruit ou lorsque les cibles sont proches. C'est pourquoi il peut s'avérer intéressant de travailler directement sur la sortie brute capteur afin d'en extraire une information plus riche. L'étape de détection créant les observations ponctuelles n'est donc plus réalisée, donnant le nom à cette nouvelle approche de track before detect (TBD). Pour

ce faire, une modélisation plus perfectionnée du capteur ainsi que de nouveaux algorithmes sont nécessaires. En effet il va falloir être capable de formuler un modèle d'observation afin de donner l'observation prédite (non bruitée) selon les cibles en présence. Dans le cas des observations ponctuelles, les mesures prédites sont souvent simples à décrire car s'expriment dans le même espace que l'espace d'état, ou un bien un espace réduit de l'espace état (par exemple, seule la position est observée, pas la vitesse), ou dans un autre système de coordonnées (par exemple en polaire plutôt qu'en cartésien). Dans le cas des données brutes, il s'agit de savoir représenter le signal de sortie du capteur. L'utilisation de données brutes permet également de s'affranchir de la problématique d'association de données entre cibles et mesures ponctuelles.

2.2.1 Méthodes initiales par batch

Pour des cibles à faible rapport signal sur bruit, il est donc difficile sur une observation brute de pouvoir repérer les cibles. En revanche, c'est l'intégration temporelle de l'information des observations brutes qui va permettre de repérer les cibles suivant leur dynamique a priori. C'est donc l'analyse sur un temps étendu avec plusieurs images d'observations brutes qui va permettre de mettre en évidence et de détecter des trajectoires de cibles en présence. Initialement, des algorithmes par lot (batch) ont donc été mis en place pour l'approche TBD, utilisant la transformée de Hough [CEW94], le maximum de vraisemblance avec fluctuation de l'amplitude de la cible [TBS98], l'algorithme de Viterbi (programmation dynamique) [Bar85]. Ces méthodes requièrent la discrétisation de l'espace d'état, ce qui permet de traiter les cas avec des équations non linéaires (comme par exemple l'équation de mesure de l'approche TBD en radar). Cependant travailler avec un espace d'état discret nécessite des ressources en mémoire et de calcul assez importantes, y compris dans des zones de l'espace d'état dans lesquelles aucune cible n'est présente et qui sont peu informatives.

2.2.2 Méthodes récursives

Pour résoudre les équations de filtrage bayésien dans un cadre TBD, les premiers travaux publiés font état de l'utilisation de filtres particuliers [NS97, BD01, SB01] afin de traiter la forte non linéarité introduite par les modèles d'observation.

Dans l'article [VVPS10], les auteurs adaptent aux observations rencontrées en TBD les équations pour trois algorithmes faisant appel au cadre des RFS. Pour cela, ils font l'hypothèse forte selon laquelle les zones d'influence des cibles sur l'image d'observation ne se chevauchent

pas. Cette hypothèse permet de dériver les expressions exactes sans passer par les approximations faites dans le cas de mesures ponctuelles. Les deux premiers cas sont ceux du filtre PHD, avec une prior Poisson, et l'autre une prior cluster i.i.d. Le troisième cas est celui du filtre Multi-Bernoulli. Une implémentation particulière de ce dernier est proposée, le modèle d'observation étant fortement non linéaire. Cependant ces méthodes ont de fortes limitations. L'hypothèse de non chevauchement ne permet pas de traiter les situations où les cibles se croisent. Le chevauchement des observations étant écarté, cela revient à découper l'espace pour chaque cible, et il serait envisageable d'avoir un filtre monocible par cible. Les auteurs concèdent d'ailleurs que l'implémentation SMC est équivalente à l'exécution en parallèle de plusieurs filtres particuliers (monocibles).

Les capteurs, ou mesures, dits sommateurs (superpositional sensors or measurements) sont un type de capteurs et mesures pour lesquels la sortie est une fonction de la somme des contributions individuelles de chaque source. Ce cas a été traité par plusieurs articles dans le cadre RFS [MEF12, NCM13, PK15]. Dans l'article [MEF12], les auteurs développent une approximation du filtre CPHD pour ce type de capteurs sommateurs, et l'implémentation du filtre avec un filtre particulière auxiliaire est donnée dans [NCM13], dans le cas d'un bruit d'observation additif et gaussien. Un filtre introduisant les labeled RFS est présenté dans [PK15], pour lequel est utilisé une distribution d'échantillonnage basée sur l'approximation CPHD précédemment citée.

Le filtre particulier TBD marginalisé [RGM04] introduit une variable d'existence permettant de définir si la cible est présente ou non. Plutôt que d'inclure cette variable d'existence dans le vecteur d'état, les auteurs proposent de calculer directement la probabilité d'existence, et les particules représentant l'état de la cible sont toutes considérées comme existantes et participent toutes à l'estimation de cet état. Les auteurs introduisent, toujours en monocible, les modèles de Swerling pour la fluctuation de l'amplitude de la cible [RGM05], et incluent alors le paramètre d'amplitude moyenne dans le vecteur d'état de la cible. En revanche, afin d'obtenir la vraisemblance à partir des modèles d'observations de Rice (en présence de la cible avec bruit) ou Rayleigh (absence de la cible et bruit seul), ils réalisent une intégration numérique pour marginaliser les distributions sur l'amplitude de la cible. McDonald et Balaji [MB08b] définissent quant à eux des expressions explicites dans le cas où les observations sont indépendantes cellule à cellule.

Dans [UMGFG17], les auteurs proposent un filtre particulier multicible en utilisant la méthode du filtre particulier auxiliaire (auxiliary particle filter, APF), pour un nombre connu et fixe de cibles. Le filtre particulier faisant appel à un processus d'échantillonnage qui s'appuie sur une densité d'importance, deux stratégies sont proposées. L'APF imite la manière dont les

échantillons seraient produits avec la densité d'importance optimale, en prenant en compte la nouvelle observation grâce à une variable auxiliaire. Le problème du MTT reste le fléau de la dimension, lorsque l'on cherche à échantillonner dans un espace à grande dimension (ce qui est le cas en MTT). Pour remédier à cela, une stratégie consiste à considérer que les états courants des cibles sont conditionnellement indépendants sachant les observations, c'est-à-dire à faire l'hypothèse d'indépendance postérieure qui s'écrit :

$$p(X_{t+1}|z_{0:t+1}) \propto p(z_{t+1}|X_{t+1}) \prod_{j=1}^N \sum_{i=1}^{N_p} w_t^i p(x_{t+1,j}|x_{t,j}^{iPP})$$

S'ajoute à cela le choix d'une densité d'importance judicieuse pour l'étape d'échantillonnage du filtre particulaire. Les filtres particuliers à partition indépendante (independent partition, IP) et à partition parallèle (parallel partition, PP) font l'hypothèse d'indépendance. Leur différence réside ensuite dans le choix de la densité d'importance. Alors que le premier (IP) ne considère pas les cibles voisines lors l'échantillonnage individuel d'une cible, le second (PP) fait intervenir les cibles voisines (via une estimation de ces dernières). C'est la seconde stratégie qui est mise en place dans [UMGFG17]. Leur approche se caractérise donc par un échantillonnage des nouvelles particules au niveau de chaque cible individuelle, en tenant compte des cibles voisines (stratégie PP), plutôt que l'état multicible directement, et orienté par la nouvelle observation (via l'APF). Un rééchantillonnage supplémentaire au niveau de chaque cible individuelle est également proposé mais pas systématique. Un seuil adaptatif est défini à chaque étape pour définir si le rééchantillonnage est à réaliser ou non.

2.2.3 Sur données réelles

Travailler sur des données simulées a permis d'éprouver les algorithmes développés dans le cadre TBD. Les données réelles représentent un défi, puisque de la bonne connaissance du modèle d'observation dépend fortement la performance de l'algorithme, et ces modèles d'observation sont souvent complexes à caractériser et déterminer explicitement.

L'article [SCSC16] présente des essais sur données réelles dans le domaine de la bathymétrie, c'est-à-dire dans la construction de carte des fonds marins. Un sonar récolte des données brutes à l'aide d'une antenne réseau et d'un faisceau étroit, permettant ainsi de définir la direction d'arrivée (Direction of Arrival, DOA) de différents signaux reflétés par les fonds marins ou d'autres objets présents. L'estimation des directions d'arrivée est menée à l'aide d'un filtre CPHD (cardinalized probability hypothesis density filter) implémenté à l'aide de méthodes SMC. Avec les données brutes, les auteurs arrivent à reconstituer la carte de fond d'un canal

peu profond dont le fond est relativement plat, avec de meilleurs résultats que de précédentes méthodes. Ils arrivent également à identifier sur une autre portion le fond du canal ainsi que la présence d'une forme cylindrique au fond du canal.

En acoustique sur données brutes, des simulations ont été menées sur des données réelles dans les articles [FG10] et [FG12] afin de localiser et pister une ou plusieurs sources sonores. Un réseau de microphones a été installé afin de détecter et suivre une ou plusieurs personnes parlant dans une pièce. Aucune détection d'activité de la voix n'est menée sur les données. Une variable d'activité de la source est ajoutée à l'état afin de définir si une source (un interlocuteur) parle (est actif) ou non. A l'aide de 12 microphones, une carte d'intensité avec des pixels est créée formant les données nécessaires au contexte TBD. Une vraisemblance spécifique au problème est alors définie et les auteurs utilisent des méthodes de filtrage particulière.

Concernant les données radar, McDonald et Balaji présentent dans [MB08b] les résultats d'un pistage sur données réelles radar dans le cadre TBD avec une seule cible manœuvrante (hors-bord) en environnement maritime. A partir de leurs enregistrements, ils caractérisent une première fois la distribution des valeurs des observations. La distribution de Rayleigh, de variance égale à celle des données réelles, semble être un choix limité car elle ne représente pas fidèlement la longue queue de la distribution des données réelles due à une plus forte probabilité d'observer des amplitudes élevées du signal correspondant à des pics de mer. En effet, d'après d'autres études, la K-distribution semble être plus représentative de la distribution des valeurs issues d'observations en mer. Cependant le choix d'une distribution de Rayleigh pour représenter les observations permet de déterminer des expressions explicites des fonctions de vraisemblances dans les cas où la cible a un modèle de fluctuation de Swerling 0, 1 ou 3. Ces mêmes auteurs comparent alors dans un autre papier [MB08a] l'utilisation des distributions de Rayleigh, de la K-distribution et la KA-distribution sur données simulées et réelles, en considérant une cible non fluctuante (modèle de Swerling 0, amplitude constante). Les données simulées sont générées à partir des caractéristiques des données réelles et selon les trois modèles de distributions de clutter cités ci-avant. Ils constatent de bonnes performances de pistage lorsque la mise à jour du filtre correspond parfaitement au modèle employé pour le jeu de données simulées. Ensuite, en appliquant les trois modèles de mise à jour (c'est-à-dire le choix de la vraisemblance) aux données simulées selon la KA-distribution, la mise à jour selon la distribution de Rayleigh présente de moins bonnes performances, selon la K-distribution de meilleures performances, et selon la KA-distribution les meilleures performances des trois, puisqu'il s'agit du modèle correspondant parfaitement au jeu de données simulées utilisé. En revanche, ces résultats ne sont pas retrouvés sur l'application aux données réelles. En effet lors de l'application sur les données simulées, il y a une adéquation parfaite entre les données (idéalisées) et le modèle utilisé, donnant d'excellents résultats. Cependant les données réelles

confrontent le modèle à un comportement plus réaliste, et le moindre écart par rapport au modèle, trop finement caractérisé, peut mener à une forte dégradation des performances. Afin de comprendre ces écarts, les données réelles sont plus finement caractérisées [MB11], au delà de leur distribution, mais également selon des schémas locaux. En effet, les retours de mer (sea clutter) ont une durée certes limitée mais non instantanée, et peuvent couvrir plusieurs cellules de résolutions de l'image d'observation, créant alors une dépendance spatio-temporelle, concrètement une "tâche" locale sur l'image due au traitement radar en azimuth et en distance. Ces études successives témoignent de la sensibilité des modèles de mise à jour aux données réelles.

Chapitre 3

Etude de filtres particuliers multicibles

Ce chapitre présente l'étude de trois stratégies de filtrage particulière dans un contexte adapté au multicible. Les trois stratégies approchent la distribution de filtrage multicible (postérieure) à l'aide d'une représentation particulière. La première stratégie, illustrée par le filtre particulier IPMCMC (Interacting Population-based Markov Chain Monte Carlo), repose sur un filtrage particulier utilisant des méthodes de Monte Carlo par chaînes de Markov afin de générer les échantillons séquentiels multicibles. Les deuxième et troisième stratégies sont des adaptations du filtrage particulier auxiliaire, et font apparaître dans ce contexte multicible un brassage des particules, appelé aussi *crossover*. La deuxième stratégie, illustrée par le filtre particulier ATRAPP (Adaptive Target Resampling Auxiliary Parallel Partition) propose d'exploiter l'hypothèse d'indépendance postérieure des cibles afin de procéder à un traitement cible par cible tout en prenant en compte les autres cibles. Cela permet d'introduire un rééchantillonnage (adaptatif) au niveau de chacune des cibles, et ce traitement cible par cible génère du *crossover*. La troisième stratégie s'affranchit de l'hypothèse d'indépendance postérieure des cibles, l'échantillonnage auxiliaire et le *crossover* étant introduits par des matrices et des densités de transition auxiliaires.

3.1 Filtre particulier IPMCMC (Interacting Population-based Markov Chain Monte Carlo particle filter)

Un algorithme de pistage multi-cibles est présenté, d'après les travaux introduits et développés dans [BDB12], dans le cas où la cardinalité de la population est fixée et connue. Il s'agit de

l’Interacting Population-based Markov Chain Monte Carlo Particle Filter (IP-MCMC-PF). Des modifications et améliorations exploitant des méthodes sur les chaînes de Markov lui sont apportées dans [IBMD15, IBD14]. Cet algorithme est étendu au cas où la cardinalité est variable et inconnue dans [BDB13], à l’aide des méthodes de Monte Carlo par chaîne de Markov à saut réversible (Reversible Jump Markov Chain Monte Carlo) présentées dans [Gre95]. Cette extension de l’IP-MCMC-PF est nommée Interacting Population-based Reversible Jump Markov Chain Monte Carlo Particle Filter (IP-RJ-MCMC-PF).

L’algorithme IP-MCMC-PF est une implémentation directe du filtre bayésien exprimé dans le cadre multi-cibles à l’aide de son approximation par les méthodes de Monte Carlo. À ce principe de méthode Monte Carlo séquentielle (SMC) est associé le principe de méthode de Monte Carlo par chaîne de Markov (Markov Chain Monte Carlo, MCMC) pour former les méthodes dites SMCMC (Sequential MCMC) ou PMCMC (Particle MCMC). Cet algorithme se différencie donc d’un filtre particulaire classique dans le sens où il ne fait pas appel à des méthodes par échantillonnage d’importance, avec ou sans rééchantillonnage, afin de construire la densité multi-cibles *a posteriori*, mais utilise des méthodes par chaînes de Markov, en s’inspirant notamment de la méthode décrite dans [KBD05].

Dans un premier temps, le nombre de cibles étant fixé et connu, les densités de probabilité sont directement exprimées sur l’espace d’état joint de toutes les cibles. Le vecteur aléatoire étudié et estimé est en conséquence la concaténation des états de toutes les cibles.

Principe du filtre particulaire IPMCMC

Cet algorithme [BDB12] utilise une méthode MCMC comme vu précédemment afin d’estimer à chaque instant t la densité de probabilité jointe sur l’état des cibles. L’idée de cet algorithme repose sur l’utilisation en parallèle de plusieurs algorithmes de Metropolis-Hastings. En effet les méthodes de filtrage particulaire étant coûteuses en temps de calcul, la parallélisation du traitement peut s’avérer efficace afin de diminuer le temps d’exécution. De plus, en simulant plusieurs chaînes de Markov simultanément, on peut espérer explorer différentes régions à forte probabilité.

Le nombre de cibles est considéré comme connu et fixé. Le vecteur d’état multi-cibles X_t représente la concaténation des états des N cibles de l’instant t

$$X_t = (x_{t,1}, \dots, x_{t,N})$$

avec $x_{t,j}$ le vecteur état de la cible j à l'instant t .

A chaque instant, la densité de probabilité multi-cibles est représentée par un ensemble de N_p particules équiprobables $\{X_t^i\}_{i=1}^{N_p}$. Le fait qu'elles soient équiprobables, c'est-à-dire que leur poids soit égal à $\frac{1}{N_p}$, vient de la construction des algorithmes MCMC comme vu ci-dessus. Pour construire la densité de probabilité à l'instant t , on possède la distribution *a priori* à l'instant $t-1$ représentée par l'ensemble de particules $\{X_{t-1}^i\}_{i=1}^{N_p}$ et une nouvelle observation z_t . Chaque particule est un vecteur contenant les états joints de chacune des cibles

$$X_t^i = (x_{t,1}^i, \dots, x_{t,N}^i)$$

avec $x_{t,j}^i$ le vecteur état de la cible j de la particule i à l'instant t .

Ainsi, l'approximation de Monte Carlo du filtre bayésien optimal, donnée par les équations (1.7) et (1.8), est la suivante

$$p(X_t|z_{0:t}) \approx c g(z_t|X_t) \frac{1}{N_p} \sum_{i=1}^{N_p} p(X_t|X_{t-1}^i) \quad (3.1)$$

avec c la constante de normalisation.

L'idée, exposée sur la figure 3.1, est alors d'utiliser, à chaque instant t , M algorithmes de Metropolis-Hastings en parallèle, chacun générant, après une période de burn-in, N_p/M échantillons, et dont la densité invariante vers laquelle les chaînes de Markov convergent est la densité de filtrage multi-cibles $p(X_t|z_{0:t})$ à l'instant t .

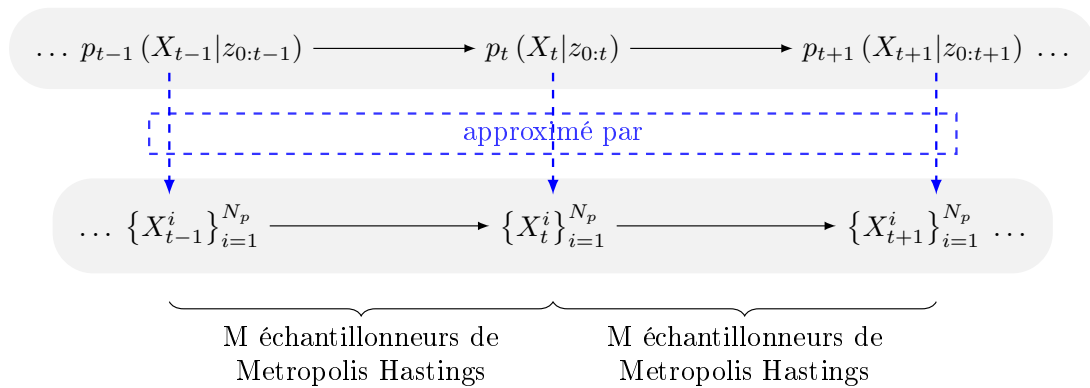


Figure 3.1: Récursion et approximation particulaire

Afin d'explorer l'espace joint multi-cible, la densité de proposition $q(X, X') = q(X_t^{s-1}, X_t^*)$

choisie remplace l'état d'une seule cible j à chaque étape s de l'algorithme MH. La densité de proposition sélectionne de façon uniforme une cible j , dont l'état va être perturbé grâce à une densité mono-cible notée $q_j(x_j, x'_j) = q_j(x_{t,j}^{s-1}, x_{t,j}^*)$, laissant les autres cibles inchangées. Pour explorer l'espace, le nouvel état proposé $x_{t,j}^*$ de la cible j ne dépend pas directement de $x_{t,j}^{s-1}$, mais est tiré selon $p(x_{t,j}|x_{t-1,j}^i)$ le modèle dynamique d'une cible, et $x_{t-1,j}^i$ l'état de la cible j d'une particule i tiré parmi la distribution *a priori* $\{X_{t-1}^i\}_{i=1}^{N_p}$ au temps $t-1$. En s'appuyant de cette manière sur les particules du temps précédent, on génère de la diversité dans les propositions pour l'algorithme MH. La figure 3.2 illustre cette transition, et la densité de proposition s'écrit alors:

$$q(X_t^{s-1}, X_t^*) = \sum_{j=1}^N \frac{1}{N} \underbrace{q_j(x_{t,j}^{s-1}, x_{t,j}^*)}_{\text{cible } j \text{ perturbée}} \underbrace{\delta_{x_{t,\neq j}^{s-1}}(x_{t,\neq j}^*)}_{\text{autres cibles inchangées}} \quad (3.2)$$

avec $x_{t,\neq j}$ le vecteur contenant les états des cibles autres que j : $x_{t,\neq j} = (x_{t,1}, \dots, x_{t,j-1}, x_{t,j+1}, \dots, x_{t,N})$ et

$$q_j(x_{t,j}^{s-1}, x_{t,j}^*) = \sum_{i=1}^{N_p} \frac{1}{N_p} p(x_{t,j}^* | x_{t-1,j}^i) \quad (3.3)$$

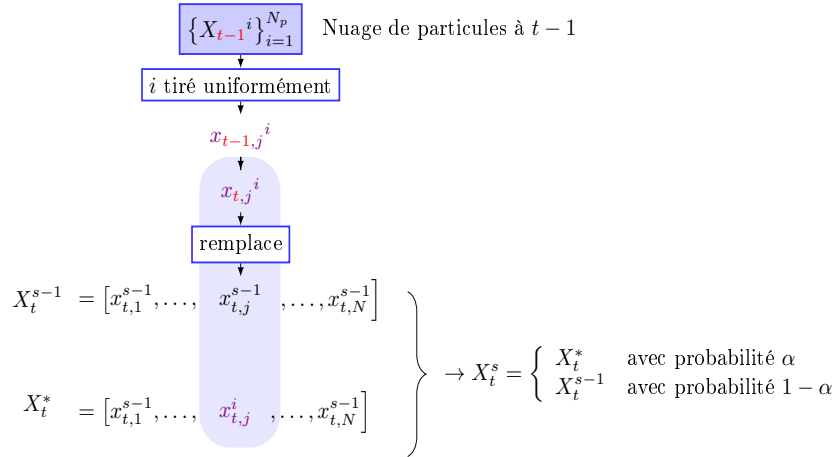


Figure 3.2: Schéma de la transition MH proposée

A l'étape s de l'échantillonnage de MH, si la cible j^* est celle tirée pour être remplacée,

alors l'évaluation en j^* de la densité de proposition (3.2), avec (3.3), donne

$$\begin{aligned} q(X_t^{s-1}, X_t^*) &= \frac{1}{N} q_{j^*}(x_{t,j^*}^{s-1}, x_{t,j^*}^*) \\ &= \frac{1}{N} \sum_{i=1}^{N_p} \frac{1}{N_p} p(x_{t,j^*}^* | x_{t-1,j^*}^i) \end{aligned} \quad (3.4)$$

La probabilité d'acceptation de l'échantillonneur de Metropolis-Hastings peut alors être déterminée avec (3.1) et (3.4)

$$\alpha(X_t^{s-1}, X_t^*) = \min \left(1, \underbrace{\frac{p(X_t^* | z_{0:t}) q(X_t^*, X_t^{s-1})}{p(X_t^{s-1} | z_{0:t}) q(X_t^{s-1}, X_t^*)}}_r \right)$$

et avec

$$\begin{aligned} r &= \frac{g(z_t | X_t^*) \left(\sum_{i=1}^{N_p} p(x_{t,j^*}^* | x_{t-1,j^*}^i) \prod_{j \neq j^*} p(x_{t,j}^* | x_{t-1,j}^i) \right) \sum_{i=1}^{N_p} p(x_{t,j^*}^{s-1} | x_{t-1,j^*}^i)}{g(z_t | X_t^{s-1}) \left(\sum_{i=1}^{N_p} p(x_{t,j^*}^{s-1} | x_{t-1,j^*}^i) \prod_{j \neq j^*} p(x_{t,j}^{s-1} | x_{t-1,j}^i) \right) \sum_{i=1}^{N_p} p(x_{t,j^*}^* | x_{t-1,j^*}^i)} \\ &= \frac{g(z_t | X_t^*) \left(\sum_{i=1}^{N_p} \alpha_{j^*}^*(i) \beta_{j^*}(i) \right) \sum_{i=1}^{N_p} \alpha_{j^*}^{s-1}(i)}{g(z_t | X_t^{s-1}) \left(\sum_{i=1}^{N_p} \alpha_{j^*}^{s-1}(i) \beta_{j^*}(i) \right) \sum_{i=1}^{N_p} \alpha_{j^*}^*(i)} \end{aligned}$$

avec

$$\begin{aligned} \alpha_{j^*}^*(i) &= p(x_{t,j^*}^* | x_{t-1,j^*}^i) \\ \alpha_{j^*}^{s-1}(i) &= p(x_{t,j^*}^{s-1} | x_{t-1,j^*}^i) \\ \beta_{j^*}(i) &= \prod_{j \neq j^*} p(x_{t,j}^* | x_{t-1,j}^i) = \prod_{j \neq j^*} p(x_{t,j}^{s-1} | x_{t-1,j}^i) \end{aligned}$$

Chaque échantillonneur MH est initialisé avec une "graine" X_t^0 obtenue en propageant à l'aide du modèle dynamique une particule tirée de la distribution *a priori* $\{X_t^i\}_{i=1}^{N_p}$. Le schéma du principe de l'algorithme IP-MCMC-PF au temps t est présenté figure 3.3 et figure 3.4.

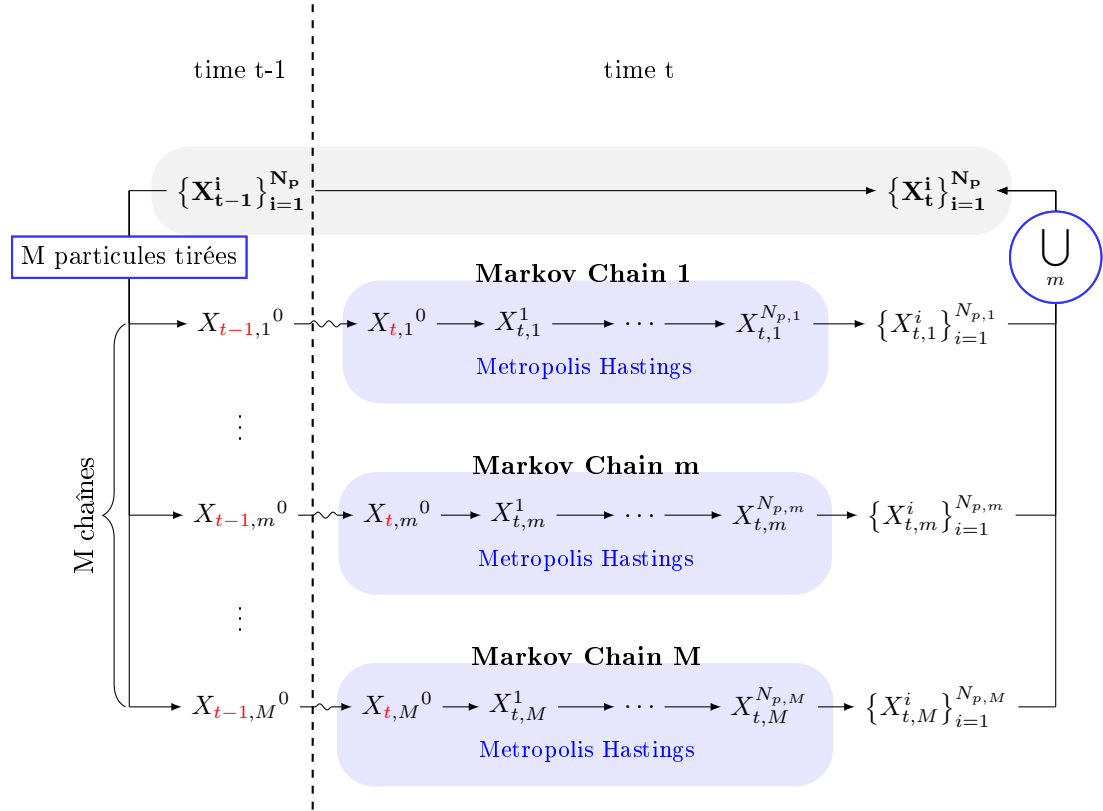


Figure 3.3: Principe du filtre IP-MCMC-PF

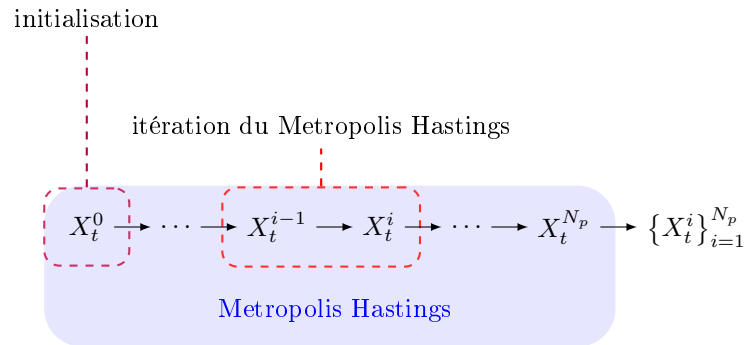


Figure 3.4: Schéma d'un Metropolis-Hastings du IP-MCMC-PF

Extension: filtre particulaire IP Reversible Jump MCMC

Afin de gérer un nombre de cible variable et inconnu, une extension du précédent filtre est proposée dans [BDB13], s'appuyant sur des méthodes de Monte Carlo par chaînes de Markov à saut réversible [Gre95, KBD05]. Ce principe autorise la dimension de l'état à varier pour chacune des particules en fonction du nombre de cibles déclarées actives. Il est donc désormais nécessaire de propager un ensemble d'indices noté $\mathcal{I}_t$ des cibles déclarées actives. L'état multicible augmenté par l'ensemble d'indices est noté $X_{\mathcal{I},t} = (X_t, \mathcal{I}_t)$. Ainsi la densité de filtrage correspondante s'écrit alors

$$p(X_{\mathcal{I},t}|z_{0:t}) \propto g(z_t|X_{\mathcal{I},t}) \times \sum_{\mathcal{I}_{t-1}} \int p(X_{\mathcal{I},t}|X_{\mathcal{I},t-1})p(X_{\mathcal{I},t-1}|z_{0:t-1})dX_{t-1}$$

et dont l'approximation particulaire, donnée à l'aide d'un nuage de particules de même poids, est définie par

$$p(X_{\mathcal{I},t}|z_{0:t}) \approx c g(z_t|X_{\mathcal{I},t}) \frac{1}{N_p} \sum_{i=1}^{N_p} f(X_{\mathcal{I},t}|X_{\mathcal{I},t-1}^i)$$

avec c la constante de normalisation.

Afin d'échantillonner les nouvelles particules, un algorithme de Metropolis-Hastings est mis en place comme précédemment. C'est la forme du noyau de proposition de la chaîne de Markov qui va permettre de changer la dimension de l'état de la particule proposée en activant une cible (naissance, apparition), en désactivant une cible (mort, disparition), ou en gardant le même nombre de cibles actives (simple mise à jour). Ces propositions sont appelés des mouvements réversibles et forment l'ensemble des mouvements possibles $\nu = \{\text{apparition(a)}, \text{disparition(d)}, \text{mise à jour (maj)}\}$. Ils sont réversibles dans la mesure où pour chaque mouvement ν , un mouvement ν_{rev} à l'effet inverse est défini. Ainsi les mouvements de mort et naissance sont réversibles l'un par rapport à l'autre, et le mouvement de mise à jour est auto-réversible (réversible par rapport à lui-même). Un mouvement qui change la dimension de l'état est appelé un saut. Il a été choisi de ne faire apparaître ou disparaître qu'une seule cible à la fois. Un mouvement ν peut être choisi avec une probabilité p_ν , et avec $\sum_\nu p_\nu = 1$. Une fois le choix du mouvement effectué, un nouvel échantillon va pouvoir être proposé, à l'étape s de la chaîne de Markov, d'après le noyau de proposition correspondant noté $q_\nu(X_{\mathcal{I},t}^{s-1}, X_{\mathcal{I},t}^*)$.

L'étape s de l'algorithme de Metropolis-Hastings va ainsi inclure le choix de mouvement permettant le changement de dimension de l'espace d'état :

1. Choix du mouvement ν d'après le vecteur de probabilité $\{p_\nu\}_\nu$
2. Proposition d'un nouvel échantillon $X_{\mathcal{I},t}^*$ selon le mouvement défini
3. Acceptation ou rejet de l'échantillon proposé

Remarque: L'apparition d'une nouvelle cible demande une initialisation de son état.

3.2 Filtre particulaire ATRAPP (Adaptive Target Resampling Auxiliary Parallel Partition particle filter)

Principe et adaptation de méthodes auxiliaire

Le principe de l'algorithme *adaptive target resampling auxiliary parallel partition particle filter* (ATRAPP) [UMFG17] fait appel à des approximations et une stratégie inspirée du filtre particulaire auxiliaire. Le terme partition désigne l'état d'une cible individuelle, considéré comme une partition de l'état multicible joint.

Le filtre particulaire constitue une approximation de la densité souhaitée. De façon séquentielle, à chaque pas de temps, la nouvelle densité à approcher est exprimée donc grâce aux équations du filtre bayésien, et à l'approximation particulaire de la densité au temps précédent. Cette expression approchée est donc l'objectif, dont on veut produire des échantillons, qui serviront pour approcher la densité au temps d'après, et ainsi de suite. Cette densité à approcher peut encore être manipulée, afin de lui donner une forme dont on considère qu'elle a de bonnes propriétés: c'est ce qui est fait lorsque l'hypothèse d'indépendance postérieure des cibles est prise. Selon cette hypothèse, une nouvelle densité postérieure est prise pour cible.

Sur l'idée du filtre particulaire auxiliaire, la densité objectif devient une densité augmentée par une variable auxiliaire. Cette variable auxiliaire est ici classiquement l'indice de la particule à partir de laquelle est échantillonné le nouvel état au temps d'après. Dans sa forme la plus classique, cet indice a une seule dimension, l'état multicible viendrait donc d'une seule particule ancêtre. Lorsque l'hypothèse d'indépendance postérieure est formulée, la structure obtenue autorise un indice différent pour chaque cible, la variable auxiliaire devient donc un vecteur de dimension le nombre de cibles. En s'appuyant sur cette variable auxiliaire, il s'agit d'un échantillonnage joint sur l'espace multicible et l'espace des indices. Les échantillons de la variable auxiliaire sont alors simplement écartés et les échantillons de l'état multicible sont conservés.

Quelle que soit la densité postérieure objectif choisie, et en considérant ou non la variable auxiliaire, il est en revanche bien souvent impossible d'échantillonner directement selon cette densité, et une densité d'importance doit donc être choisie. Assez naturellement, la densité

d'importance va souvent avoir une forme assez proche de la densité objectif mais avec des solutions permettant le tirage d'échantillons. Donc si l'hypothèse d'indépendance des cibles a été formulée pour la densité objectif, on peut imaginer une forme similaire pour la densité d'importance. Certaines méthodes échantillonnent les cibles indépendamment les unes des autres, tandis que d'autres méthodes échantillonnent chacune des cibles en prenant en compte les autres cibles (ou les plus proches) tout en gardant l'hypothèse d'indépendance postérieure [YMKY13].

La structure de la densité d'importance à partir de laquelle l'état multicible est échantillonné permet de réaliser un échantillonnage cible par cible (échantillonneur indépendant), donc pouvant être fait en parallèle, et en prenant compte d'une certaine manière les autres cibles, donnant le terme *parallel partition* dans le nom du filtre.

L'idée de ce filtre est le rééchantillonnage par cible adaptatif (*adaptive target resampling*), et peut être approchée selon deux angles. Tout d'abord, le rééchantillonnage adaptatif est préconisé dans l'utilisation d'un filtre particulaire afin d'éviter la dégénérescence des poids et l'appauvrissement des échantillons. Mais plutôt que de l'appliquer à l'échelle des particules multicible, comme l'échantillonnage est réalisé cible par cible, il pourrait être avantageux de faire le rééchantillonnage à l'échelle des particules monocibles. Cela est possible en considérant donc deux étapes dans une boucle monocible. La première étape est l'échantillonnage des particules monocibles grâce à une densité d'importance selon laquelle il est possible d'échantillonner. La seconde étape est alors le possible rééchantillonnage, possible car adaptatif, selon un critère qui doit être défini. Un critère bien connu pour le rééchantillonnage est le nombre d'échantillons effectifs, utilisé dans le cadre d'un échantillonnage d'importance. Donc pour avoir accès à un tel critère, une densité monocible intermédiaire est introduite. Ainsi le poids des particules tirées selon la première densité monocible d'importance peut être calculé et le nombre de particules effectif associé déterminé. Si le nombre de particules effectif est supérieur au seuil prédéfini, alors il n'y a pas de rééchantillonnage et les particules sont bien distribuées selon la première densité d'importance monocible. Si le nombre de particules effectif est inférieur au seuil prédéfini, les particules sont rééchantillonnées selon les poids précédemment calculés et elles sont donc désormais distribuées selon la densité intermédiaire qui devient alors la densité d'importance utilisée. Cette démarche étant adoptée pour le traitement de chaque cible indépendamment, certaines cibles auront eu un rééchantillonnage de leurs particules, tandis que d'autres non, donnant donc au final une densité d'importance multicible hybride entre les densités d'importance monocibles selon lesquelles les échantillons ont pu être tirés (pour les cibles n'ayant pas eu de rééchantillonnage) et les densités monocibles intermédiaires (pour les cibles ayant été rééchantillonnées).

Hypothèse d'indépendance postérieure et expressions

Considérant l'ensemble de particules pondérées au temps précédent $\{w_{t-1}^i, X_{t-1}^i\}_{i=1}^{N_p}$, avec l'état multicible $X_t = [x_{t,1}, \dots, x_{t,N}]$, l'approximation particulaire de la densité de filtrage au temps $t - 1$ s'écrit donc

$$p(X_{t-1}|z_{0:t-1}) \approx \sum_{i=1}^{N_p} w_{t-1}^i \delta_{X_{t-1}^i}(X_{t-1})$$

Cette approximation de la densité *a priori* peut donc être injectée dans les équations du filtre bayésien afin de donner la densité de filtrage multicible postérieure (la densité objectif selon laquelle l'échantillonnage de particules est souhaité)

$$p(X_t|z_{0:t}) \tilde{\propto} p(z_t|X_t) \sum_{i=1}^{N_p} w_{t-1}^i p(X_t|X_{t-1}^i) \quad (3.5)$$

Les cibles évoluent indépendamment dynamiquement, ce qui se traduit par

$$p(X_t|X_{t-1}^i) = \prod_{j=1}^N p(x_{t,j}|x_{t-1,j}^i) \quad (3.6)$$

et l'expression (3.5) peut alors se réécrire

$$p(X_t|z_{0:t}) \tilde{\propto} p(z_t|X_t) \sum_{i=1}^{N_p} w_{t-1}^i \prod_{j=1}^N p(x_{t,j}|x_{t-1,j}^i)$$

Sous cette forme, toutes les cibles d'un nouvel échantillon multicible seront donc issues de la même particule ancêtre.

Si on fait l'hypothèse d'indépendance des cibles sur la densité de filtrage postérieure au temps $t - 1$ alors l'application de cette hypothèse sur l'approximation particulaire de cette densité s'écrit comme suit

$$p(X_{t-1}|z_{0:t-1}) \approx \prod_{j=1}^N \left[\sum_{a_j=1}^{N_p} w_{t-1}^{a_j} \delta_{x_{t-1,j}^{a_j}}(x_{t-1,j}) \right] \quad (3.7)$$

Cette expression (3.7), couplée avec la remarque sur l'indépendance dynamique des cibles

(3.6), donne alors l'expression pour la densité postérieure multicible suivante

$$p(X_t|z_{0:t}) \propto p(z_t|X_t) \prod_{j=1}^N \sum_{a_j=1}^{N_p} w_{t-1}^{a_j} p(x_{t,j}|x_{t-1,j}^{a_j})$$

Cette approximation construite à partir de l'approximation particulaire au temps précédent sous l'hypothèse d'indépendance postérieure des cibles, est la densité multicible à partir de laquelle les particules sont à échantillonner. Sous cette forme, les cibles d'un nouvel échantillon multicible pourront être issues de particules ancêtres différentes.

Afin de faciliter l'échantillonnage, cette densité est augmentée avec une variable auxiliaire. Dans le filtre particulaire auxiliaire classique cette variable est l'indice de la particule ancêtre à partir de laquelle est tirée le nouvel échantillon. Mais cette variable auxiliaire peut avoir toute autre forme, et ici elle prend la forme d'un vecteur d'indice des particules ancêtres dont viennent chacune des cibles du nouvel échantillon. La densité augmentée, ou jointe, avec la variable auxiliaire s'exprime alors de la façon suivante

$$p(X_t, a|z_{0:t}) \propto p(z_t|X_t) \prod_{j=1}^N w_{t-1}^{a_j} p(x_{t,j}|x_{t-1,j}^{a_j}) \quad (3.8)$$

avec $a = [a_1, \dots, a_N] \in \{1, \dots, N_p\}^N$ le vecteur des indices des particules ancêtres dont sont issues chacune des cibles du nouvel échantillon. Une fois la variable auxiliaire échantillonnée sur laquelle peut alors s'appuyer l'échantillonnage de l'état multicible d'intérêt, la variable auxiliaire n'est plus considérée et abandonnée pour la suite du filtrage.

L'expression (3.8) est donc la densité objectif selon laquelle de nouvelles particules sont souhaitées. Bien que la variable auxiliaire vienne aider l'échantillonnage, il n'est pas possible d'échantillonner directement à partir de cette densité, une densité d'importance $q(X_t, a|z_{1:t})$ doit être élaborée afin d'y parvenir. Les particules multicibles échantillonnées sont alors pondérées de la façon suivante

$$w_t^i = \frac{p(X_t^i, a^i|z_{0:t})}{q(X_t, a|z_{0:t})}$$

et le calcul ne nécessite de connaître les expressions explicites des deux densités qu'à une constante multiplicative près à condition de normaliser les poids obtenus.

La construction de la densité d'importance va fortement s'appuyer sur la forme de l'expression objectif.

Le schéma d'un filtre particulaire auxiliaire est souvent le suivant:

1. Echantillonnage de la variable auxiliaire
2. Echantillonnage de l'état d'intérêt connaissant la variable auxiliaire

3. Pondération de l'échantillon

Densités d'importance et rééchantillonnage individuel

Ce schéma s'applique donc afin d'obtenir le nombre souhaité de particules, les échantillons étant tirés les uns après les autres ou simultanément selon les possibilités d'implémentation.

$$q(X_t, a|z_{0:t}) = q(a|z_{0:t})q(X_t|a, z_{0:t}) \quad (3.9)$$

Le choix explicite de la forme des deux composantes de la densité d'importance formulée par (3.9) permet de la réécrire de la manière suivante

$$q(X_t, a|z_{0:t}) = \prod_{j=1}^N q(x_{t,j}, a_j|z_{0:t}) \quad (3.10)$$

avec donc au niveau de chaque cible une densité d'importance de la forme

$$q(x_{t,j}, a_j|z_{0:t}) = q(a_j|z_{0:t})q(x_{t,j}|a_j, z_{0:t})$$

La formulation de la densité d'importance multicible va plus loin dans l'indépendance des cibles comme en témoigne l'expression (3.10). Cette formulation va donc permettre d'échantillonner les cibles indépendamment les unes des autres. Or une forme de dépendance va être introduite dans la stratégie d'échantillonnage des cibles afin de prendre en compte les autres cibles. En effet la vraisemblance s'exprimant conditionnellement à l'état multicible, cette propriété est exploitée afin de tirer l'état d'une cible en prenant en compte les autres cibles.

On note $\hat{x}_{t|t-1,\ell}$ l'estimation de l'état prédit de la cible ℓ , c'est-à-dire $\hat{x}_{t|t-1,\ell} = \sum_{i=1}^{N_p} w_{t-1}^i x_{t|t-1,\ell}^i$ avec $x_{t|t-1,\ell}^i = \mathbb{E} [x_{t,\ell}|x_{t-1,\ell}^i]$ la moyenne de l'état prédit de la cible ℓ de la particule i . Ainsi est formé un vecteur des autres cibles que celle qui est échantillonnée de la manière suivante: $\hat{X}_{t,\setminus j} = [\hat{x}_{t,1}, \dots, \hat{x}_{t,j-1}, \hat{x}_{t,j+1}, \dots, \hat{x}_{t,N}]$. A la manière du filtre particulaire auxiliaire classique, l'échantillonnage de l'état de la cible va s'appuyer sur une caractéristique notée $\lambda_{t,j}^i$ de cette cible. Cette caractéristique peut être par exemple la moyenne selon la densité de transition ou un échantillon selon cette même densité de transition de la cible

$$\begin{aligned} \text{exemple 1} \quad \lambda_{t,j}^i &= \mathbb{E} [x_{t,j}|x_{t-1,j}^i] = x_{t|t-1,j}^i \\ \text{exemple 2} \quad \lambda_{t,j}^i &\sim p(x_{t,j}|x_{t-1,j}^i) \end{aligned}$$

Ainsi, une densité d'importance monocible (appelée aussi ici densité initiale), avec variable

auxiliaire et prenant en compte les autres cibles que celle échantillonnée, est définie par

$$q_{\text{init}}(x_{t,j}, a_j | z_{0:t}) \propto p(z_t | x_{t,j}, \hat{X}_{t,\setminus j}) w_{t-1}^{a_j} p(x_{t,j} | x_{t-1,j}^{a_j}) \quad (3.11)$$

Il n'est cependant pas possible d'échantillonner directement d'après cette densité d'importance et d'appliquer directement le schéma d'échantillonnage à la manière du filtre particulaire auxiliaire. Afin d'obtenir des échantillons selon cette densité d'importance initiale (3.11), une étape d'échantillonnage d'importance intermédiaire est donc réalisée. La densité intermédiaire monocible, selon laquelle il est possible d'échantillonner, et permettant donc de fournir les échantillons pour la cible j , est définie comme suit

$$q_{\text{inter}}(x_{t,j}, a_j | z_{0:t}) \propto p(z_t | \lambda_{t,j}^{a_j}, \hat{X}_{t,\setminus j}) w_{t-1}^{a_j} p(x_{t,j} | x_{t-1,j}^{a_j}) \quad (3.12)$$

La différence entre cette densité d'importance (3.12) et la précédente définie par l'expression (3.11) réside dans la substitution l'état de la cible lui-même par le paramètre caractéristique de la cible, à la manière d'un filtre particulaire auxiliaire.

Ainsi les poids intermédiaires des ces échantillons sont définis de la manière suivante

$$\begin{aligned} r_{t,j}^i &= \frac{q_{\text{init}}(x_{t,j}^i, a_j^i | z_{0:t})}{q_{\text{inter}}(x_{t,j}^i, a_j^i | z_{0:t})} \\ &\propto \frac{p(z_t | x_{t,j}^i, \hat{X}_{t,\setminus j})}{p(z_t | \lambda_{t,j}^{a_j^i}, \hat{X}_{t,\setminus j})} \end{aligned}$$

Afin d'obtenir des échantillons selon la densité d'importance initiale considérée (3.11), il est donc nécessaire de rééchantillonner à partir des poids intermédiaires, à l'échelle de la cible, c'est le *target resampling*. Or une étape de rééchantillonnage introduit toujours une réduction dans la variété des particules, puisque certaines seront abandonnées et d'autres dupliquées car ayant un poids supérieur. Afin de mitiger cet inconvénient, le rééchantillonnage n'est pas systématiquement réalisé, mais est adaptatif, et décidé selon le critère classique du nombre de particules efficaces, définit par

$$N_{\text{eff},j} = \frac{1}{\sum_{i=1}^{N_p} (r_{t,j}^i)^2}$$

Si le nombre de particules efficaces est supérieur à un certain seuil, alors le rééchantillonnage n'est pas considéré comme nécessaire et n'est donc pas effectué. Les particules restent donc échantillonnées selon la première densité de proposition. Si le nombre de particules efficaces tombe sous ce seuil, alors les particules sont rééchantillonnées selon les poids normalisés intermédiaires, et les particules sont donc désormais distribuées selon la densité de proposi-

tion intermédiaire. Cette étape introduit donc bien un rééchantillonnage à l'échelle de la cible adaptatif (*adaptive target resampling*). La densité de proposition intermédiaire, pouvant être considérée comme plus générale que la densité de proposition initiale selon laquelle il est possible d'échantillonner, sert ici d'outil pour assurer la légitimité des particules monocibles tirées selon la première densité de proposition, et proposer une étape de rééchantillonnage adaptatif à ce niveau, permettant ainsi de s'affranchir du rééchantillonnage adaptatif des particules multicibles formées ensuite.

Ainsi pour chacune des cibles, selon que le rééchantillonnage ait été déclenché ou non, les particules monocibles sont issues soit de la première densité de proposition, soit de la densité de proposition intermédiaire. Les particules monocibles sont concaténées pour former les particules multicibles. La densité de proposition multicible est donc une densité hybride, selon de quelle densité de proposition viennent les particules de chacun des cibles, et s'exprime

$$q(X_t, a|z_{0:t}) = \prod_{j/N_{\text{eff},j} \geq N_{\text{seuil}}} q_{\text{inter}}(x_{t,j}^i, a_{t,j}^i|z_{0:t}) \prod_{j/N_{\text{eff},j} < N_{\text{seuil}}} q_{\text{init}}(x_{t,j}^i, a_{t,j}^i|z_{0:t})$$

Donc finalement les poids des particules multicibles ainsi assemblées est calculé de la manière suivante

$$\begin{aligned} w_t^i &= \frac{p(X_t^i, a^i|z_{0:t})}{q(X_t^i, a^i|z_{0:t})} \\ &\propto \frac{p(z_t|X_t^i)}{\prod_{j/N_{\text{eff},j} \geq N_{\text{seuil}}} p(z_t|\lambda_{t,j}^{a_j^i}, \hat{X}_{t,\setminus j}) \prod_{j/N_{\text{eff},j} < N_{\text{seuil}}} p(z_t|x_{t,j}^i, \hat{X}_{t,\setminus j})} \end{aligned}$$

Les particules ayant déjà été rééchantillonnées à l'échelle de la cible, un rééchantillonnage des particules multicibles n'est pas réalisé. Le schéma 3.5 illustre le principe de l'algorithme.

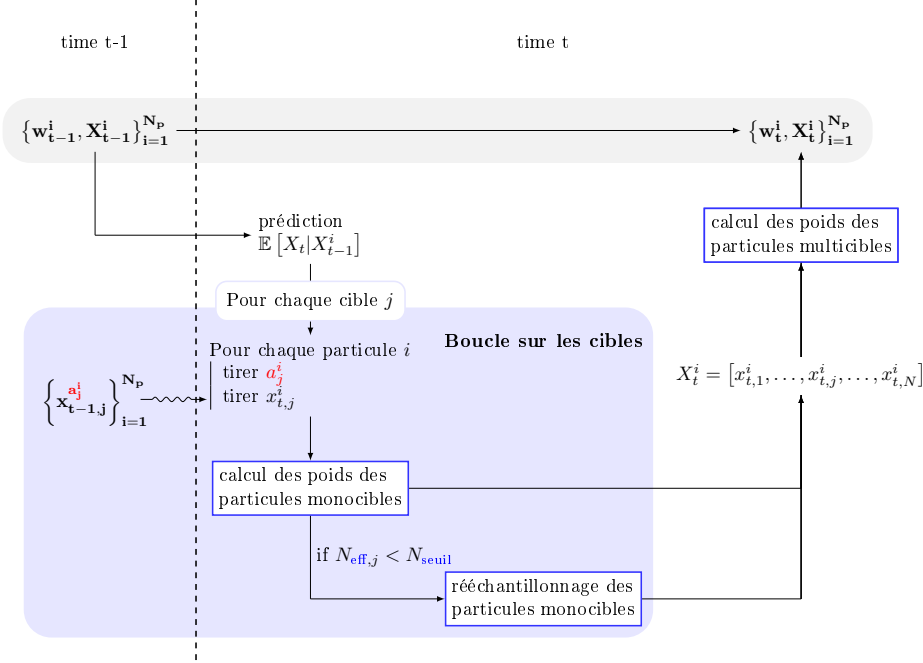


Figure 3.5: Schéma du principe de ATRAPP

3.3 Filtre particulaire avec transition markovienne auxiliaire et crossover

Dans la section précédente, et sous l'hypothèse d'indépendance *a posteriori* des cibles, il a été possible de briser l'association forcée des particules des différentes cibles au sein d'une particule multi-cible. Cette liberté offerte de brasser les particules des différentes cibles est très utile et désirable, mais l'hypothèse de départ sur laquelle elle repose est malheureusement fautive dans la plupart des applications. Concrètement, il faudrait que la fonction de vraisemblance, vue comme une fonction du vecteur multi-cible, se factorise comme un produit de fonctions ne dépendant chacune que d'une cible. Par exemple une situation avec plusieurs capteurs fonctionnant de manière indépendante (avec des bruits d'observation mutuellement indépendants) et chaque capteur fournissant une observation d'une cible seulement.

L'objectif de cette section est de proposer une approche originale, permettant aussi de brasser les particules des différentes cibles mais sans faire l'hypothèse d'indépendance *a posteriori* des cibles. Le point de départ de cette approche est l'introduction d'une transition markovienne auxiliaire, vue comme généralisation du filtre particulaire auxiliaire. Cette méthode est davantage développée et détaillée dans l'annexe A.

A l'instant t , la densité objectif selon laquelle l'échantillonnage de particules est souhaité s'écrit

$$p(X_t \mid z_{0:t}) \propto p(z_t \mid X_t) \sum_{i=1}^{N_p} w_{t-1}^i p(X_t \mid X_{t-1}^i)$$

où $X_t = (x_{t,1}, \dots, x_{t,N})$ est le vecteur d'état multi-cible, à valeurs dans l'espace d'état multi-cibles $\mathcal{X}$, et $x_{t,j}$ est le vecteur d'état de la j -ème cible pour tout $j = 1, \dots, N$, à valeurs dans l'espace d'état mono-cible $\mathcal{X}_j$, et clairement $\mathcal{X} = \mathcal{X}_1 \times \dots \times \mathcal{X}_N$. Compte tenu que les cibles évoluent indépendamment les unes des autres, la densité de transition se factorise

$$p(X_t \mid X_{t-1}^i) = \prod_{j=1}^N p(x_{t,j} \mid x_{t-1,j}^i)$$

et la densité objectif (3.3) peut alors se réécrire comme

$$p(X_t \mid z_{0:t}) \propto p(z_t \mid X_t) \sum_{i=1}^{N_p} w_{t-1}^i \prod_{j=1}^N p(x_{t,j} \mid x_{t-1,j}^i) .$$

3.3.1 Matrice de transition auxiliaire

En introduisant une matrice de transition auxiliaire $\pi_{\text{aux}} = (\pi_{\text{aux}}^{i,j_1, \dots, j_N})$ de dimension $N_p \times N_p^N$, de $\mathcal{I} = \{1, \dots, N_p\}$ vers $\mathcal{I}^N = \{1, \dots, N_p\} \times \dots \times \{1, \dots, N_p\}$ (N fois), vue comme une collection indexée par $i \in \mathcal{I}$ de vecteurs de probabilité sur $\mathcal{I}^N$, l'idée est de voir la densité objectif comme la densité marginale sur $\mathcal{X}$ de la distribution jointe

$$p_{\text{aux}}^{a, c_1, \dots, c_N}(X_t) \propto p(z_t \mid X_t) w_{t-1}^a \pi_{\text{aux}}^{a, c_1, \dots, c_N} \prod_{j=1}^N p(x_{t,j} \mid x_{t-1,j}^a)$$

définie sur l'espace produit $\mathcal{I} \times \mathcal{I}^N \times \mathcal{X}$. En effet, si on somme par rapport à l'indice $a \in \mathcal{I}$ et au multi-indice $(c_1, \dots, c_N) \in \mathcal{I}^N$, on obtient

$$\begin{aligned}
& \sum_{a=1}^{N_p} \sum_{c_1=1}^{N_p} \dots \sum_{c_N=1}^{N_p} p_{\text{aux}}^{a, c_1, \dots, c_N}(X_t) \\
& \propto p(z_t \mid X_t) \sum_{a=1}^{N_p} w_{t-1}^a \left[\sum_{c_1=1}^{N_p} \dots \sum_{c_N=1}^{N_p} \pi_{\text{aux}}^{a, c_1, \dots, c_N} \right] \prod_{j=1}^N p(x_{t,j} \mid x_{t-1,j}^a) \\
& \propto p(z_t \mid X_t) \sum_{a=1}^{N_p} w_{t-1}^a \prod_{j=1}^N p(x_{t,j} \mid x_{t-1,j}^a) \\
& \propto p(X_t \mid z_{0:t}) .
\end{aligned}$$

Il s'agit donc d'échantillonner selon la distribution jointe (3.3.1), puis de marginaliser l'approximation ainsi obtenue. En pratique, on décompose la distribution jointe comme

$$p_{\text{aux}}^{a, c_1, \dots, c_N}(X_t) \propto p(z_t \mid X_t) \underbrace{\frac{w_{t-1}^a}{\lambda_{\text{aux}}^a} \prod_{j=1}^N \frac{p(x_{t,j} \mid x_{t-1,j}^a)}{p(x_{t,j} \mid x_{t-1,j}^{c_j})}}_{r_{\text{aux}}^{a, c_1, \dots, c_N}(X_t)} \underbrace{\lambda_{\text{aux}}^a \pi_{\text{aux}}^{a, c_1, \dots, c_N} \prod_{j=1}^N p(x_{t,j} \mid x_{t-1,j}^{c_j})}_{q_{\text{aux}}^{a, c_1, \dots, c_N}(X_t)}$$

en faisant apparaître une distribution d'importance jointe et une fonction de pondération, comme dans l'échantillonnage pondéré. Clairement, les particules multi-cibles $(x_{t-1,1}^a, \dots, x_{t-1,N}^a)$ initialement présentes dans la population à l'instant $t-1$, sont possiblement remplacées par des particules multi-cibles $(x_{t-1,1}^{c_1}, \dots, x_{t-1,N}^{c_N})$ et c'est bien la matrice de transition π_{aux} qui permet ce brassage entre les particules des différentes cibles. L'approximation par échantillonnage pondéré de la distribution jointe selon la décomposition (3.3.1) peut se décrire ainsi

$$p_{\text{aux}}^{a, c_1, \dots, c_N}(X_t) \approx \sum_{i=1}^N w_t^i \delta_{(a^i, c_1^i, \dots, c_N^i, X_t^i)}$$

et sa marginale

$$p(X_t \mid z_{0:t}) \approx \sum_{i=1}^N w_t^i \delta_{X_t^i}$$

où indépendamment pour tout $i = 1, \dots, N_p$, la i -ème particule $(a^i, c_1^i, \dots, c_N^i, X_t^i)$ à valeurs dans $\mathcal{I} \times \mathcal{I}^N \times \mathcal{X}$ est échantillonnée selon la distribution d'importance jointe $q_{\text{aux}}^{a, c_1, \dots, c_N}(X_t)$ et reçoit un poids évalué grâce à la fonction de pondération $r_{\text{aux}}^{a, c_1, \dots, c_N}(X_t)$. Concrètement, indépendamment pour tout $i = 1, \dots, N_p$

- l'indice a^i est échantillonné selon le vecteur de probabilité auxiliaire $\lambda_{\text{aux}} = (\lambda_{\text{aux}}^1, \dots, \lambda_{\text{aux}}^{N_p})$,

puis, en posant $a = a^i$

- le multi-indice $(c_1^i, \dots, c_N^i)$ est échantillonné selon la a -ème ligne de la matrice de transition auxiliaire π_{aux} , c'est-à-dire selon le vecteur de probabilité $(\pi_{\text{aux}}^{a,1,\dots,1}, \dots, \pi_{\text{aux}}^{a,N_p,\dots,N_p})$,

puis, en posant $(c_1, \dots, c_N) = (c_1^i, \dots, c_N^i)$

- la particule $x_{t,j}^i$ pour la j -ème cible est échantillonnée selon la densité $p(x_{t,j} \mid x_{t-1,j}^{c_j})$, pour tout $j = 1, \dots, N$,

et finalement

- la particule multi-cible $X_t^i = (x_{t,1}^i, \dots, x_{t,N}^i)$ reçoit le poids $w_t^i \propto r_{\text{aux}}^{a,c_1,\dots,c_N}(X_t^i)$, soit

$$w_t^i \propto p(z_t \mid X_t^i) \frac{w_{t-1}^a}{\lambda_{\text{aux}}^a} \prod_{j=1}^N \frac{p(x_{t,j}^i \mid x_{t-1,j}^a)}{p(x_{t,j}^i \mid x_{t-1,j}^{c_j})}.$$

Cette classe de filtres particuliers dépend de $(N_p - 1) + (N_p^{N+1} - N_p) = N_p^{N+1} - 1 = O(N_p^{N+1})$ degrés de liberté, ou paramètres de conception, qui sont les N_p composantes du vecteur de probabilité auxiliaire λ_{aux} et les N_p lignes de la matrice de transition auxiliaire π_{aux} , c'est-à-dire N_p vecteurs de probabilité auxiliaires de dimension N_p^N , sous les contraintes de normalisation. Cette classe contient la classe des filtres particuliers auxiliaires — et a fortiori elle contient le filtre particulier ordinaire — comme cas particulier. En effet

- si pour tout $a \in \mathcal{I}$ la a -ème ligne de la matrice de transition auxiliaire π_{aux} satisfait $\pi_{\text{aux}}^{a,c_1,\dots,c_N} = 1$ si et seulement si $c_1 = \dots = c_N = a$, et $\pi_{\text{aux}}^{a,c_1,\dots,c_N} = 0$ sinon, alors le filtre particulier ainsi défini appartient à la classe des filtres particuliers auxiliaires,
- et si en outre $\lambda_{\text{aux}}^a = w_{t-1}^a$ pour tout $a \in \{1, \dots, N_p\}$, alors on retrouve le filtre particulier ordinaire.

La procédure correspondante est présentée par l'algorithme 7.

Parmi tous les choix possibles pour le vecteur de probabilité auxiliaire λ_{aux} et pour la matrice de transition auxiliaire π_{aux} , le choix qui minimise la variance des poids est défini de la façon suivante : pour tout $a \in \mathcal{I}$ on pose

$$u_a^{c_1,\dots,c_N} = \int_{\mathcal{X}_1} \dots \int_{\mathcal{X}_N} |p(z_t \mid X_t)|^2 \prod_{j=1}^N \left| \frac{p(x_{t,j} \mid x_{t-1,j}^a)}{p(x_{t,j} \mid x_{t-1,j}^{c_j})} \right|^2 \prod_{j=1}^N p(x_{t,j} \mid x_{t-1,j}^{c_j}) dx_{t,1} \dots dx_{t,N}$$

pour tout multi-indice $(c_1, \dots, c_N) \in \mathcal{I}^N$, on définit u_a^* comme le minimum de $u_a^{c_1, \dots, c_N}$ quand le multi-indice $(c_1, \dots, c_N)$ parcourt $\mathcal{I}^N$ et on suppose qu'il existe un unique multi-indice $(c_1^*(a), \dots, c_N^*(a))$ où le minimum est atteint. Alors, le choix optimal de la matrice de transition auxiliaire est tel que $\pi_{\text{opt}}^{a, c_1, \dots, c_N} = 1$ si et seulement si $(c_1, \dots, c_N) = (c_1^*(a), \dots, c_N^*(a))$ et $\pi_{\text{opt}}^{a, c_1, \dots, c_N} = 0$ sinon, et le choix optimal du vecteur de probabilité auxiliaire est

$$\lambda_{\text{opt}}^a \propto w_{t-1}^a \sqrt{u_a^*}.$$

Avec ce choix optimal des paramètres de conception, indépendamment pour tout $i = 1, \dots, N_p$

- l'indice a^i est échantillonné selon le vecteur de probabilité auxiliaire $\lambda_{\text{opt}} = (\lambda_{\text{opt}}^1, \dots, \lambda_{\text{opt}}^{N_p})$,

puis, en posant $a = a^i$ et $(c_1, \dots, c_N) = (c_1^*(a), \dots, c_N^*(a))$

- la particule $x_{t,j}^i$ pour la j -ème cible est échantillonnée selon la densité $p(x_{t,j} \mid x_{t-1,j}^{c_j})$, pour tout $j = 1, \dots, N$,

et finalement

- la particule multi-cible $X_t^i = (x_{t,1}^i, \dots, x_{t,N}^i)$ reçoit le poids

$$w_t^i \propto \frac{1}{\sqrt{u_a^*}} p(z_t \mid X_t^i) \prod_{j=1}^N \frac{p(x_{t,j}^i \mid x_{t-1,j}^{a_{t-1,j}})}{p(x_{t,j}^i \mid x_{t-1,j}^{c_j})}.$$

Le défi pour tout $a \in \mathcal{I}$ consiste à calculer

$$u_a^{c_1, \dots, c_N} = \int_{\mathcal{X}_1} \dots \int_{\mathcal{X}_N} |p(z_t \mid X_t)|^2 \prod_{j=1}^N \left| \frac{p(x_{t,j} \mid x_{t-1,j}^a)}{p(x_{t,j} \mid x_{t-1,j}^{c_j})} \right|^2 \prod_{j=1}^N p(x_{t,j} \mid x_{t-1,j}^{c_j}) dx_{t,1} \dots dx_{t,N}$$

pour tout multi-indice $(c_1, \dots, c_N) \in \mathcal{I}^N$, de façon à obtenir la valeur minimale u_a^* et le minimiseur $(c_1^*(a), \dots, c_N^*(a))$, ce qui revient à effectuer N_p recherches exhaustives parmi N_p^N éléments !

3.3.2 Densité de transition auxiliaire

En introduisant une densité de transition auxiliaire $\kappa_{\text{aux}} = (\kappa_{\text{aux}}^i(y_1, \dots, y_N))$ de $\mathcal{I} = \{1, \dots, N_p\}$ vers $\mathcal{X}$, vue comme une collection indexée par $i \in \mathcal{I}$ de densités jointes sur $\mathcal{X}$,

l'idée est de voir la densité objectif comme la densité marginale sur $\mathcal{X}$ de la distribution jointe

$$p_{\text{aux}}^a(y_1, \dots, y_N, X_t) \propto p(z_t | X_t) w_{t-1}^a \kappa_{\text{aux}}^a(y_1, \dots, y_N) \prod_{j=1}^N p(x_{t,j} | x_{t-1,j}^a)$$

définie sur l'espace produit $\mathcal{I} \times \mathcal{X} \times \mathcal{X}$. En effet, si on somme et intègre par rapport à l'indice $a \in \mathcal{I}$ et à la variable multi-cible $(y_1, \dots, y_N) \in \mathcal{X}$ respectivement, on obtient

$$\begin{aligned} & \sum_{a=1}^{N_p} \int_{\mathcal{X}_1} \dots \int_{\mathcal{X}_N} p_{\text{aux}}^a(y_1, \dots, y_N, X_t) dy_1 \dots dy_N \\ & \propto p(z_t | X_t) \sum_{a=1}^{N_p} w_{t-1}^a \left[\int_{\mathcal{X}_1} \dots \int_{\mathcal{X}_N} \kappa_{\text{aux}}^a(y_1, \dots, y_N) dy_1 \dots dy_N \right] \prod_{j=1}^N p(x_{t,j} | x_{t-1,j}^a) \\ & \propto p(z_t | X_t) \sum_{a=1}^{N_p} w_{t-1}^a \prod_{j=1}^N p(x_{t,j} | x_{t-1,j}^a) \\ & \propto p(X_t | z_{0:t}) . \end{aligned}$$

Il s'agit donc d'échantillonner selon la distribution jointe (3.3.2), puis de marginaliser l'approximation ainsi obtenue. En fait, on décompose la distribution jointe comme

$$p_{\text{aux}}^a(y_1, \dots, y_N, X_t) \propto p(z_t | X_t) \underbrace{\frac{w_{t-1}^a}{\lambda_{\text{aux}}^a} \prod_{j=1}^N \frac{p(x_{t,j} | x_{t-1,j}^a)}{p(x_{t,j} | y_j)}}_{r_{\text{aux}}^a(y_1, \dots, y_N, X_t)} \underbrace{\lambda_{\text{aux}}^a \kappa_{\text{aux}}^a(y_1, \dots, y_N) \prod_{j=1}^N p(x_{t,j} | y_j)}_{q_{\text{aux}}^a(y_1, \dots, y_N, X_t)}$$

en faisant apparaître une distribution d'importance jointe et une fonction de pondération, comme dans l'échantillonnage pondéré. Clairement, les particules multi-cibles $(x_{t-1,1}^a, \dots, x_{t-1,N}^a)$ initialement présentes dans la population à l'instant $t-1$, sont possiblement remplacées par des particules multi-cibles $(y_1, \dots, y_N)$ et c'est bien la densité de transition κ_{aux} qui permet cette redistribution des particules des différentes cibles. L'approximation par échantillonnage pondéré de la distribution jointe selon la décomposition (3.3.2) peut se décrire ainsi

$$p_{\text{aux}}^a(y_1, \dots, y_N, X_t) \approx \sum_{i=1}^{N_p} w_t^i \delta_{(a^i, y_1^i, \dots, y_N^i, X_t^i)}$$

et sa marginale

$$p(X_t | z_{0:t}) \approx \sum_{i=1}^{N_p} w_t^i \delta_{X_t^i}$$

où indépendamment pour tout $i = 1, \dots, N_p$, la i -ème particule $(a^i, y_1^i, \dots, y_N^i, X_t^i)$ à valeurs dans $\mathcal{I} \times \mathcal{X} \times \mathcal{X}$ est échantillonnée selon la distribution d'importance jointe $q_{\text{aux}}^a(y_1, \dots, y_N, X_t)$ et reçoit un poids évalué grâce à la fonction de pondération $r_{\text{aux}}^a(y_1, \dots, y_N, X_t)$. Concrètement, indépendamment pour tout $i = 1, \dots, N_p$

- l'indice a^i est échantillonné selon le vecteur de probabilité auxiliaire $\lambda_{\text{aux}} = (\lambda_{\text{aux}}^1, \dots, \lambda_{\text{aux}}^{N_p})$,

puis, en posant $a = a^i$

- le vecteur multi-cible $(y_1^i, \dots, y_N^i)$ est échantillonné selon la a -ème ligne de la densité de transition auxiliaire κ_{aux} , c'est-à-dire selon la densité jointe $\kappa_{\text{aux}}^a(y_1, \dots, y_N)$,

puis, en posant $(y_1, \dots, y_N) = (y_1^i, \dots, y_N^i)$

- la particule $x_{t,j}^i$ pour la j -ème cible est échantillonnée selon la densité $p(x_{t,j} | y_j)$, pour tout $j = 1, \dots, N$,

et finalement

- la particule multi-cible $X_t^i = (x_{t,1}^i, \dots, x_{t,N}^i)$ reçoit le poids $w_t^i \propto r_{\text{aux}}^a(y_1, \dots, y_N, X_t^i)$, soit

$$w_t^i \propto p(z_t | X_t^i) \frac{w_{t-1}^a}{\lambda_{\text{aux}}^a} \prod_{j=1}^N \frac{p(x_{t,j}^i | x_{t-1,j}^a)}{p(x_{t,j}^i | y_j)}.$$

Cette classe de filtres particuliers dépend de plusieurs paramètres de conception, qui sont les N_p composantes du vecteur de probabilité auxiliaire λ_{aux} et les N_p densités auxiliaires définies sur $\mathcal{X}$. Cette classe contient la classe des filtres particuliers avec matrice de transition auxiliaire — et a fortiori elle contient la classe des filtres particuliers auxiliaires et le filtre particulier ordinaire — comme cas particulier. En effet

- si pour tout $a \in \mathcal{I}$ la densité auxiliaire $\kappa_{\text{aux}}^a(y_1, \dots, y_N)$ s'exprime comme un mélange fini

$$\kappa_{\text{aux}}^a(y_1, \dots, y_N) = \sum_{c_1=1}^{N_p} \dots \sum_{c_N=1}^{N_p} \pi_{\text{aux}}^{a,c_1,\dots,c_N} \prod_{j=1}^N \delta_{x_{t-1,j}^{c_j}}(y_j)$$

où le vecteur des poids de mélange est la a -ème ligne d'une matrice de transition π_{aux} de dimension $N_p \times N_p^N$, alors la densité auxiliaire $\kappa_{\text{aux}}^a(y_1, \dots, y_N)$ permet juste d'échantillonner le vecteur multi-cible parmi l'ensemble des vecteurs de la forme $(x_{t-1,1}^{c_1}, \dots, x_{t-1,N}^{c_N})$ et on vérifie que le filtre particulier ainsi défini appartient à la classe des filtres particuliers avec matrice de transition auxiliaire, décrit à la section 3.3.1,

- si en outre pour tout $a \in \mathcal{I}$ la a -ème ligne de la matrice de transition auxiliaire π_{aux} satisfait $\pi_{\text{aux}}^{a, c_1, \dots, c_N} = 1$ si et seulement si $c_1 = \dots = c_N = a$, et $\pi_{\text{aux}}^{a, c_1, \dots, c_N} = 0$ sinon, alors la densité auxiliaire prend une forme dégénérée (concentrée sur un unique vecteur multi-cible)

$$\kappa_{\text{aux}}^a(y_1, \dots, y_N) = \prod_{j=1}^N \delta_{x_{t-1,j}^a}(y_j) = \delta_{(x_{t-1,1}^a, \dots, x_{t-1,N}^a)}(y_1, \dots, y_N)$$

ce qui fixe l'état multi-cible comme la a -ème particule multi-cible $(x_{t-1,1}^a, \dots, x_{t-1,N}^a)$, et on vérifie que le filtre particulaire ainsi défini appartient à la classe des filtres particuliers auxiliaires,

- et si en outre pour tout $a \in \mathcal{I}$ on pose $\lambda_{\text{aux}}^a = w_{t-1}^a$, alors on retrouve le filtre particulaire ordinaire.

La procédure correspondante est présentée par l'algorithme 8.

Parmi tous les choix possibles pour le vecteur de probabilité auxiliaire λ_{aux} et pour la densité de transition auxiliaire κ_{aux} , le choix qui minimise la variance des poids est défini de la façon suivante : pour tout $a \in \mathcal{I}$ on pose

$$u_a(y_1, \dots, y_N) = \int_{\mathcal{X}_1} \dots \int_{\mathcal{X}_N} |p(z_t | X_t)|^2 \prod_{j=1}^N \left| \frac{p(x_{t,j} | x_{t-1,j}^a)}{p(x_{t,j} | y_j)} \right|^2 \prod_{j=1}^N p(x_{t,j} | y_j) dx_{t,1} \dots dx_{t,N}$$

pour tout vecteur multi-cible $(y_1, \dots, y_N) \in \mathcal{X}$, on définit u_a^* comme le minimum de la fonction $(y_1, \dots, y_N) \mapsto u_a(y_1, \dots, y_N)$ quand le vecteur multi-cible $(y_1, \dots, y_N)$ parcourt $\mathcal{X}$ et on suppose qu'il existe un unique vecteur multi-cible $(y_1^*(a), \dots, y_N^*(a))$ où le minimum est atteint. Alors, le choix optimal de la densité de transition auxiliaire est

$$\kappa_{\text{opt}}^a(y_1, \dots, y_N) = \delta_{(y_1^*(a), \dots, y_N^*(a))}(y_1, \dots, y_N)$$

et le choix optimal du vecteur de probabilité auxiliaire est

$$\lambda_{\text{opt}}^a \propto w_{t-1}^a \sqrt{u_a^*}.$$

Avec ce choix optimal des paramètres de conception, indépendamment pour tout $i = 1, \dots, N_p$

- l'indice a^i est échantillonné selon le vecteur de probabilité auxiliaire $\lambda_{\text{opt}} = (\lambda_{\text{opt}}^1, \dots, \lambda_{\text{opt}}^{N_p})$,

puis, en posant $a = a^i$ et $(y_1, \dots, y_N) = (y_1^*(a), \dots, y_N^*(a))$

- la particule $x_{t,j}^i$ pour la j -ème cible est échantillonnée selon la densité $p(x_{t,j} \mid y_j)$, pour tout $j = 1, \dots, N$,

et finalement

- la particule multi-cible $X_t^i = (x_{t,1}^i, \dots, x_{t,N}^i)$ reçoit le poids

$$w_t^i \propto \frac{1}{\sqrt{u_a^*}} p(z_t \mid X_t^i) \prod_{j=1}^N \frac{p(x_{t,j}^i \mid x_{t-1,j}^a)}{p(x_{t,j}^i \mid y_j)}.$$

Le défi pour tout $a \in \mathcal{I}$ consiste à calculer

$$u_a(y_1, \dots, y_N) = \int_{\mathcal{X}_1} \dots \int_{\mathcal{X}_N} |p(z_t \mid X_t)|^2 \prod_{j=1}^N \left| \frac{p(x_{t,j} \mid x_{t-1,j}^a)}{p(x_{t,j} \mid y_j)} \right|^2 \prod_{j=1}^N p(x_{t,j} \mid y_j) dx_{t,1} \dots dx_{t,N}$$

pour tout vecteur multi-cible $(y_1, \dots, y_N) \in \mathcal{X}$, de façon à obtenir la valeur minimale u_a^* et le minimiseur $(y_1^*(a), \dots, y_N^*(a))$, ce qui revient à résoudre numériquement N_p problèmes d'optimisation dans un espace de dimension $N \times \dim \mathcal{X}$.

Algorithme 7 : Filtre particulaire avec matrice de transition auxiliaire

Input : $z_{0:t}$
Output : $\{w_{0:T}^i, X_{0:T}^i\}_{i=1}^{N_p}$
 $A \ t = 0$
Initialisation
for $i = 1$ *to* N_p **do**
 Echantillonner $X_0^i \sim q_0(\cdot)$
 Calculer le poids non normalisé $\tilde{w}_0^i = p(z_0|X_0^i) \frac{p(X_0^i)}{q_0(X_0^i)}$
end
Normaliser le poids $w_0^i = \frac{\tilde{w}_0^i}{\sum_{i=1}^{N_p} \tilde{w}_0^i} \quad \forall i = 1, \dots, N_p$
 $t = 1$
while $t \leq T$ **do**
 1. Echantillonnage de la variable auxiliaire
 for $i = 1$ *to* N_p **do**
 Echantillonner a^i selon le vecteur de probabilité auxiliaire
 $\lambda_{\text{aux}} = (\lambda_{\text{aux}}^1, \dots, \lambda_{\text{aux}}^{N_p})$
 end
 2. Echantillonnage du multi-indice du crossover
 for $i = 1$ *to* N_p **do**
 Poser $a = a^i$
 Echantillonner $(c_1^i, \dots, c_N^i)$ selon la a -ème ligne de la matrice de transition auxiliaire π_{aux} , c'est-à-dire selon le vecteur de probabilité $(\pi_{\text{aux}}^{a,1,\dots,1}, \dots, \pi_{\text{aux}}^{a,N_p,\dots,N_p})$
 end
 3. Propagation / Echantillonnage des états
 for $i = 1$ *to* N_p **do**
 Poser $a = a^i$ et $(c_1, \dots, c_N) = (c_1^i, \dots, c_N^i)$
 for $j = 1$ *to* N **do**
 Echantillonner $x_{t,j}^i \sim p(x_{t,j}|x_{t-1,j}^{c_j})$
 end
 4. Pondération
 Calculer le poids non normalisé de la particule multi-cible
 $\tilde{w}_t^i = p(z_t|X_t^i) \frac{w_{t-1}^a}{\lambda_{\text{aux}}^a} \prod_{j=1}^N \frac{p(x_{t,j}^i|x_{t-1,j}^a)}{p(x_{t,j}^i|x_{t-1,j}^{c_j})}$
 end
 Normaliser les poids $w_t^i = \frac{\tilde{w}_t^i}{\sum_{i=1}^{N_p} \tilde{w}_t^i} \quad \forall i = 1, \dots, N_p$
 $t = t + 1$
end

Algorithme 8 : Filtre particulaire avec densité de transition auxiliaire**Input** : $z_{0:t}$ **Output** : $\{w_{0:T}^i, X_{0:T}^i\}_{i=1}^{N_p}$ $A \ t = 0$ Initialisation**for** $i = 1$ *to* N_p **do**| Echantillonner $X_0^i \sim q_0(\cdot)$ | Calculer le poids non normalisé $\tilde{w}_0^i = p(z_0|X_0^i) \frac{p(X_0^i)}{q_0(X_0^i)}$ **end**Normaliser le poids $w_0^i = \frac{\tilde{w}_0^i}{\sum_{i=1}^{N_p} \tilde{w}_0^i} \quad \forall i = 1, \dots, N_p$ $t = 1$ **while** $t \leq T$ **do**1. Echantillonnage de la variable auxiliaire**for** $i = 1$ *to* N_p **do**| Echantillonner a^i selon le vecteur de probabilité auxiliaire| $\lambda_{\text{aux}} = (\lambda_{\text{aux}}^1, \dots, \lambda_{\text{aux}}^{N_p})$ **end**2. Echantillonnage du vecteur multi-cible du crossover**for** $i = 1$ *to* N_p **do**| Poser $a = a^i$ | Echantillonner $(y_1^i, \dots, y_N^i)$ selon la a -ème ligne de la densité de transition auxiliaire κ_{aux} , c'est-à-dire selon la densité jointe $\kappa_{\text{aux}}^a(y_1, \dots, y_N)$ **end**3. Propagation / Echantillonnage des états**for** $i = 1$ *to* N_p **do**| Poser $a = a^i$ et $(y_1, \dots, y_N) = (y_1^i, \dots, y_N^i)$ **for** $j = 1$ *to* N **do**| Echantillonner $x_{t,j}^i \sim p(x_{t,j}|y_j)$ **end**4. Pondération

Calculer le poids non normalisé de la particule multi-cible

$$\tilde{w}_t^i = p(z_t|X_t^i) \frac{w_{t-1}^i}{\lambda_{\text{aux}}^a} \prod_{j=1}^N \frac{p(x_{t,j}^i|x_{t-1,j}^a)}{p(x_{t,j}^i|y_j)}$$

endNormaliser les poids $w_t^i = \frac{\tilde{w}_t^i}{\sum_{i=1}^{N_p} \tilde{w}_t^i} \quad \forall i = 1, \dots, N_p$ $t = t + 1$ **end**

Chapitre 4

Simulations et résultats

4.1 Avant-propos

Afin d'éprouver les algorithmes précédemment présentés et d'établir leurs performances, des simulations sont menées. Ces simulations font appels aux modélisations dynamique et d'observation également présentées dans ce manuscrit. Les deux algorithmes multicibles présentés cherchent à approcher la même densité postérieure multicible à l'aide de méthodes particulières. L'algorithme [ATRAPP](#) cependant admet une hypothèse supplémentaire sur l'approximation particulière de cette densité postérieure multicible: l'hypothèse d'indépendance postérieure des cibles.

Pour ces deux algorithmes, les méthodes retenues afin d'échantillonner les particules sont différentes:

- le filtre particulière [IPMCMC](#) utilise une méthode [MCMC](#),
- le filtre particulière [ATRAPP](#) utilise une méthode de type filtre particulière auxiliaire.

Présence de *crossover*

Ces deux approches permettent à leur manière de réaliser un *crossover* entre les particules, en mélangeant dans les particules multicibles des cibles issues de différentes particules. Dans le filtre [ATRAPP](#), chaque cible d'une nouvelle particule multicible est issue d'une particule précédente différente (particule ancêtre ou auxiliaire), permettant un mélange des particules (le *crossover*) important. Le filtre [IPMCMC](#) réalise un *crossover* plus doux. En effet la chaîne de

Markov propose une particule multicible dont une cible à la fois est issue d'une autre particule (les autres cibles restant inchangées). Il est donc intéressant de confronter ces deux types de *crossover*.

Variantes de l'ATRAPP

Sont aussi testés les algorithmes [Auxiliary Parallel Partition \(APP\)](#) et [Target Resampling Auxiliary Parallel Partition \(TRAPP\)](#). Ces deux algorithmes sont des variantes de l'algorithme [ATRAPP](#), dans lesquels le rééchantillonnage par cible est soit exclu soit obligatoire, ce qui se traduit par un choix de seuil de rééchantillonnage sur chaque cible:

- dans l'[APP](#), il n'y a pas de rééchantillonnage par cible, c'est-à-dire que le seuil de rééchantillonnage est fixé à zéro, $N_{\text{seuil}} = 0$. Il n'est donc pas nécessaire de calculer les poids intermédiaires, ce qui est un gain de calculs. Le *crossover* est bien réalisé à travers l'échantillonnage des variables auxiliaires et des particules ancêtres associées.
- dans le [TRAPP](#), le rééchantillonnage par cible est systématique pour chaque cible, c'est-à-dire que le seuil de rééchantillonnage est fixé à un, $N_{\text{seuil}} = 1$. Le rééchantillonnage n'est pas adaptatif mais est forcé à chaque fois. De même les variables auxiliaires sont toujours échantillonnées au préalable, ce qui permet le *crossover*.

Coût de calcul

Le coût de calcul augmente avec le nombre de cibles et le nombre de particules. Le calcul de la vraisemblance est assez complexe, ne peut être vectorisé et représente donc la plus grande charge de calculs. Chacun des algorithmes fait appel à de multiples calculs de la vraisemblance, que ce soit dans le filtre [ATRAPP](#) pour la pondération auxiliaire, le rééchantillonnage des cibles et la pondération des particules multicibles, ou bien dans le filtre [IPMCMC](#) pour le ratio d'acceptation dans chaque chaîne de Markov. Le nombre de calculs de la vraisemblance pour chacun des algorithmes est présenté dans la table 4.1. Pour le filtre [ATRAPP](#), ce nombre dépend directement du nombre de cibles N et du nombre de particules N_p , tandis que pour le filtre [IPMCMC](#), il dépend du nombre de particules N_p ainsi que du nombre N_{MCMC} de chaînes de Markov lancées en parallèle et du nombre B de particules de la période de *burn-in*, mais pas du nombre de cibles. Cependant la taille de l'espace d'état multicible dépend du nombre de cibles, et influence le nombre N_{MCMC} de chaînes de Markov et le nombre B du *burn-in* nécessaires afin d'explorer correctement tout l'espace d'état multicible afin d'atteindre la distribution stationnaire, puisque la chaîne explore l'espace cible après cible.

ATRAPP TRAPP	APP	IPMCMC
$2 \times N \times N_p + N_p$	$N \times N_p + N_p$	$(N_{p,\text{MCMC}} + B) \times N_{\text{MCMC}}$ $= N_p + B \times N_{\text{MCMC}}$

N : nombre de cibles

N_p : nombre de particules

N_{MCMC} : nombre de chaînes de Markov

$N_{p,\text{MCMC}}$: nombre de particules par chaîne de Markov

B : nombre de particules du *burn-in*

Table 4.1: Nombre de calculs de vraisemblance

4.2 Modèle de Swerling 3

Le scénario présenté ici comporte 3 cibles se déplaçant selon un modèle à vitesse constante, et suivant un modèle de Swerling 3 pour l'observation. Les algorithmes quant à eux utilisent le modèle dynamique à vitesse quasi constante, et également le modèle de Swerling 3, qui est connu, pour l'amplitude des cibles. La paramètre d'amplitude des cibles est également considéré comme connu, et peut être déterminé à chaque instant connaissant le [signal-to-noise ratio \(SNR\)](#), ou rapport signal à bruit en français, et la position de la cible. Le scénario est présenté en figure 4.1.

Le champ de vision (*field of view*) du radar, ainsi que la durée et la pas de temps choisis pour ce scénario sont définis dans la table 4.2.

Champ de vision radar	distance	[2000 m, 4000 m]
	résolution en distance	$\Delta_r = 10$ m
	angle	$[-10^\circ, +10^\circ]$
	résolution en angle	$\Delta_\theta = 1^\circ$
Temps	durée	60 s
	intervalle de temps	$\Delta_t = 2$ s

Table 4.2: Paramètres du radar et de durée

Les cibles sont initialisées de telle sorte qu'elles ne sortent pas du champ de vision durant la simulation. Les valeurs d'initialisation sont disponibles dans la table 4.3, avec r_0 la distance de la cible au capteur, θ_0 l'azimut, $v_0 = (\dot{x} \ \dot{y})_{t=0}$ le vecteur vitesse initial. Tous les angles ont

pour direction de référence l'axe vertical nord.

	r_0	θ_0	$ v_0 $	$\hat{v}_0$	SNR (à 3000m)
Cible 1	3400 m	-5°	18 m.s $^{-1}$	170°	10 dB
Cible 2	3200 m	0°	8 m.s $^{-1}$	20°	10 dB
Cible 3	3000 m	5°	10 m.s $^{-1}$	-175°	10 dB

Table 4.3: Valeurs d'initialisation des cibles

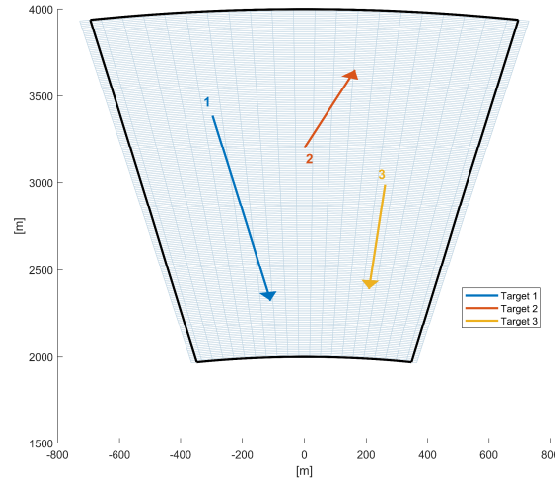


Figure 4.1: Scénario à 3 cibles

La puissance du signal réfléchi par la cible est d'autant plus faible que la cible est éloignée, et d'autant plus forte que la cible est proche. Le SNR fluctue donc au cours du temps selon que la cible s'éloigne ou s'approche du capteur. Chaque cible a un SNR égal à 10dB à une distance de référence de 3000m du capteur, correspondant au milieu du champ de vision. Les SNR de chaque cible en début et fin de scénario sont présentés dans la table 4.4. Les paramètres des distributions des amplitudes sont considérés comme connus puisque directement liés aux SNR des cibles par la relation définie par:

$$\text{SNR} = 20 \log \left(\frac{a}{\sigma r^2} \right)$$

avec a l'amplitude, σ l'écart-type du bruit d'observation et r la distance (de référence).

Tous les paramètres utilisés dans les filtres sont disponibles dans la table 4.5. Les résultats sont présentés sur 20 réalisations de Monte Carlo, sur le même jeu de données. Le nuage

	SNR $t = 0$	SNR $t = \text{fin}$
Cible 1	7.8	14.4
Cible 2	8.9	6.6
Cible 3	10	13.9

Table 4.4: Evolution du SNR des cibles

de particules est initialisé autour des positions réelles des cibles, à l'aide d'une distribution gaussienne. Les figures 4.2a et 4.2b illustrent chacune une réalisation du filtrage particulaire pour chacun des filtres sur les composantes de position des cibles.

	ATRAPP	IPMCMC
N_p	500	
N_{MCMC}		10
B		100
Calculs vraisemblance (Table 4.1)	3500	1500
Seuil de rééchantillonnage N_{seuil}	0.5	
Variance dynamique $\sigma_x^2 = \sigma_y^2$		0.1
Variance du bruit σ_n^2		1

Table 4.5: Paramètres des filtres

Sur la figure 4.2, il semble que l'incertitude du filtre ATRAPP est plus élevée que celle du filtre IPMCMC. Cette remarque doit être modérée par le fait que le filtre ATRAPP présente des particules pondérées, tandis que le filtre IPMCMC propage des particules de poids égal.

Pour évaluer les algorithmes, la racine de l'erreur quadratique moyenne, ou **Root Mean Square Error (RMSE)**, est calculée en position et en vitesse pour chacune des cibles. Cette erreur est donnée à partir de l'estimation d'état des filtres (moyenne pondérée des particules), et de l'état réel des cibles. Les résultats des filtres sont présentés sur la figure 4.3 pour la position, et sur la figure 4.4 pour la vitesse.

Sur ces graphiques, les courbes du filtre ATRAPP et du filtre TRAPP se chevauchent en position. L'algorithme IPMCMC est plus performant que l'ATRAPP et ses variantes, en étant plus robuste et présentant une erreur plus lisse. Bien qu'étant plus sensible et variable face

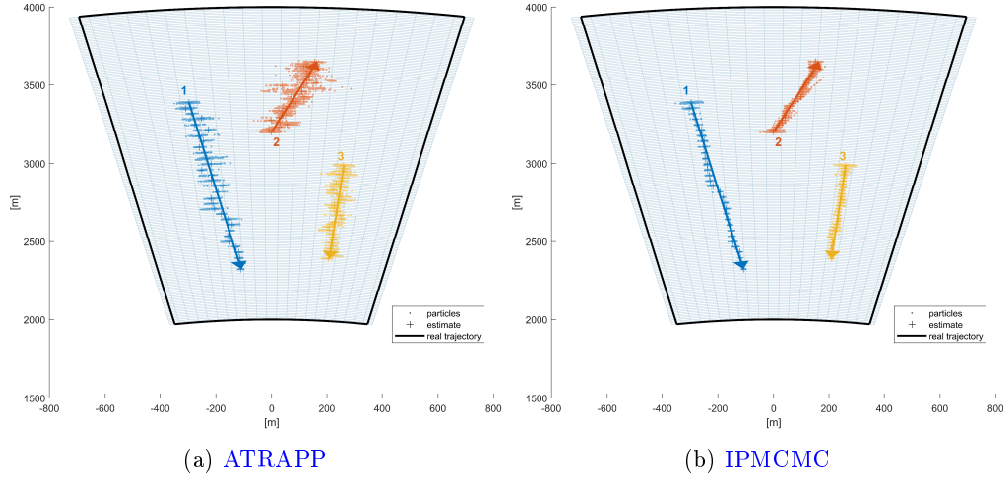


Figure 4.2: Résultats des filtres particulaires sur une réalisation

aux observations, le filtre **ATRAPP** produit un résultat similaire en position au filtre **IPMCMC** sur certains pas de temps, ce qui est encourageant, et indique qu'il ne diverge pas. Le filtre **IPMCMC** est autorisé à dupliquer les particules (grâce à la méthode d'acceptation et rejet du Metropolis-Hastings), lui permettant lorsqu'il converge vers une zone intéressante de l'espace d'y rester, au prix d'une diversité réduite parmi les particules. De son côté, le filtre **ATRAPP** adapte son rééchantillonnage si nécessaire, permettant de garder une meilleure diversité dans ses particules. Le filtre **APP** donnent des résultats moins lisse que l'**IPMCMC** mais le concurrence très régulièrement. Ce scénario ne proposant pas de croisement ou de fort rapprochement entre les cibles ne permet pas de mettre en valeur l'intérêt du rééchantillonnage par cible proposé par les filtres **ATRAPP** et **TRAPP**, tandis que la variante **APP** profite bien du l'échantillonnage auxiliaire.

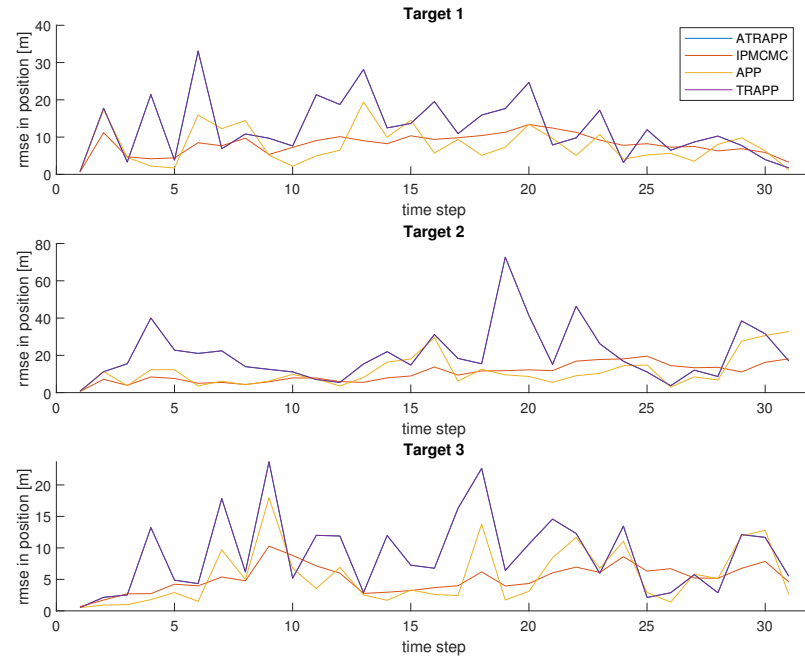


Figure 4.3: RMSE en positions

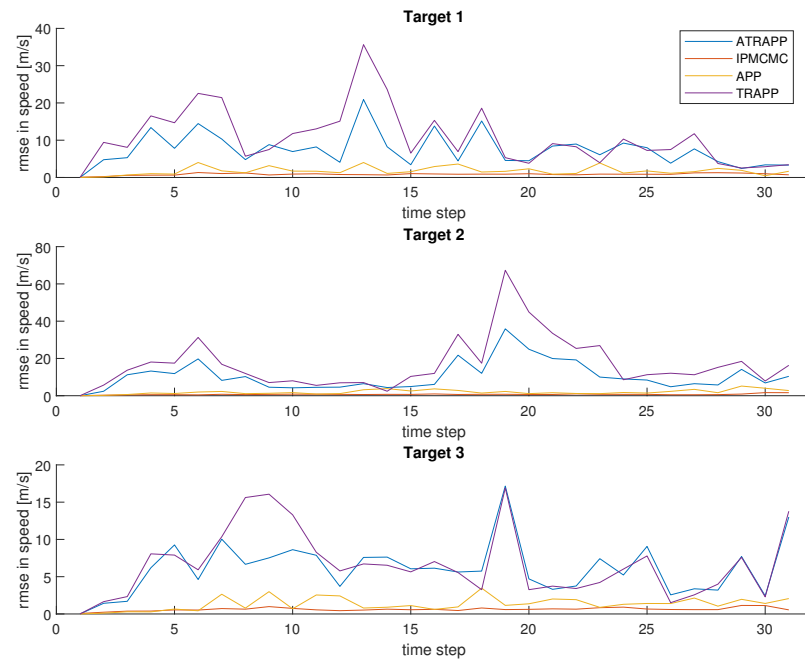


Figure 4.4: RMSE en vitesses

4.3 Modèle de Swerling 0

Le précédent scénario permet d'avoir un premier aperçu des performances avec une gestion multi-cible, mais dans un contexte où les cibles sont éloignées et n'interagissent pas entre elles. Dans ce deuxième scénario, des croisements entre les cibles sont observés afin d'éprouver les méthodes de suivi multi-cible. Le scénario comporte ainsi 5 cibles. Chacune des cibles suit ici un modèle de Swerling 0, donc l'amplitude des cibles est connue, et de même liée au SNR de chacune des cibles. Le scénario est présenté par la figure 4.5, les valeurs d'initialisation ainsi que l'évolution du SNR de chacune des cibles sont présentés à la table 4.6. Les trajectoires des cibles se croisent. Les distances entre les couples de cibles qui se croisent sont illustrées par la figure 4.6, afin de rendre compte de leur éloignement au cours du temps. Les paramètres du radar et de durée du scénario sont les mêmes que dans la table 4.2.

	r_0	θ_0	$ v_0 $	$\hat{v}_0$	SNR (à 3000m)	SNR $t = 0$	SNR $t = \text{fin}$
Cible 1	3400 m	-5°	18 m.s ⁻¹	155°	10 dB	7.8	13.8
Cible 2	3200 m	0°	8 m.s ⁻¹	20°	10 dB	8.9	6.6
Cible 3	3000 m	5°	12 m.s ⁻¹	-155°	10 dB	10	14.3
Cible 4	2700 m	-8°	10 m.s ⁻¹	95°	8 dB	9.8	10.3
Cible 5	3500 m	-4°	14 m.s ⁻¹	100°	13 dB	10.3	10.8

Table 4.6: Valeurs d'initialisation des cibles et évolution du SNR

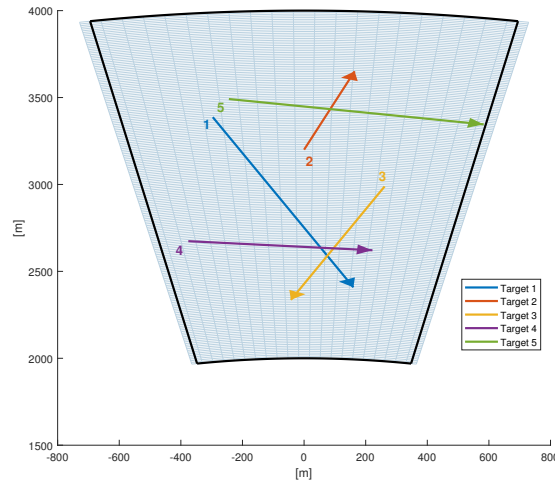


Figure 4.5: Scénario à 5 cibles

Les résultats sont présentés sur 100 réalisations de Monte Carlo, sur des jeux de données différents pour chaque réalisation. Le nuage de particule est initialisé autour des positions réelles

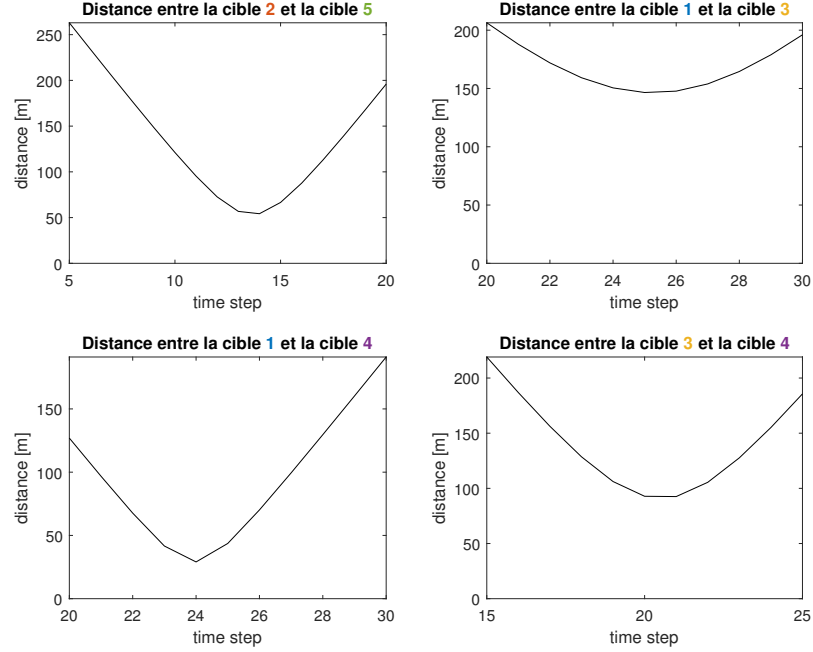


Figure 4.6: Distances entre cibles

des cibles, à l'aide d'une distribution gaussienne. Les figures 4.7a et 4.7b illustrent chacune une réalisation du filtrage particulaire pour chacun des filtres sur les composantes de position des cibles.

Les figures 4.8 et 4.9 présentent les résultats des filtres en positions et en vitesses. De même que précédemment c'est l'erreur quadratique moyenne, ou **RMSE**, qui permet de comparer ici les performances des algorithmes.

Dans ce scénario dans lequel les cibles sont amenées à être proches et à se croiser, le filtre **APP** est moins performant que les trois autres filtres. En effet le filtre **APP** n'a pas d'étape de rééchantillonnage au niveau des cibles, les échantillons pour la cible j sont tirés selon la densité (3.12). Sans cette étape de rééchantillonnage des particules au niveau de la cible, le graphique 4.9 de la **RMSE** en vitesse illustre la mauvaise performance du filtre **APP** sur la composante vitesse par rapport aux trois autres algorithmes. Plus le nombre de cibles est élevé plus l'espace d'état est grand. Le rééchantillonnage permet alors d'éviter la dégénérescence des particules, notamment dans le cas d'un espace d'état de grande dimension. Dans ce scénario à 5 cibles, la performance de l'algorithme **TRAPP**, qui inclut un rééchantillonnage systématique au niveau des cibles, obtient une meilleure performance que l'algorithme **APP**, montrant ainsi la nécessité

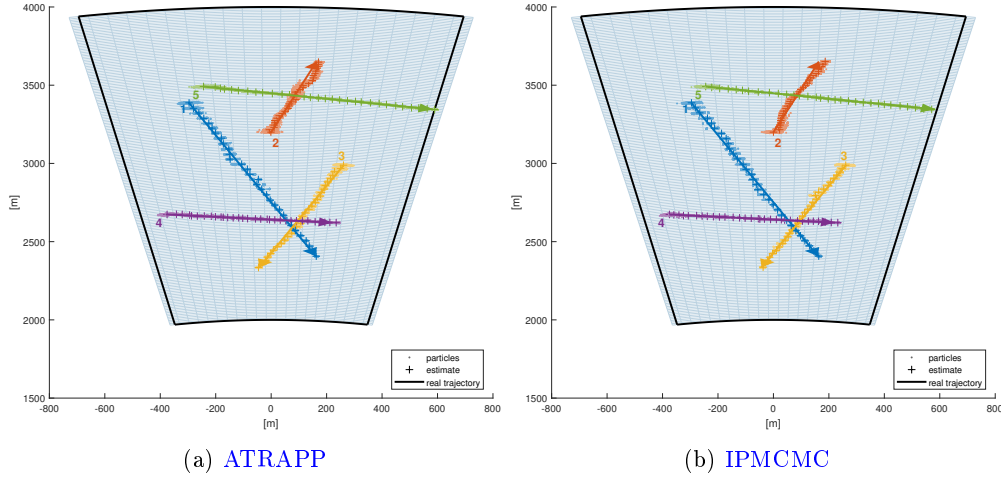


Figure 4.7: Résultats des filtres particulaires sur une réalisation

du rééchantillonnage au niveau des cibles dans ce scénario. L'échantillonnage des variables auxiliaires peut être vu comme un rééchantillonnage des particules sur les particules ancêtres, évitant tout de même une dégénérescence complète de l'algorithme, c'est-à-dire qu'une seule particule ait tout le poids (un poids proche de un) et les autres particules des poids presque nuls.

Les algorithmes TRAPP et ATRAPP présentent de meilleures performances en position que l'algorithme IPMCMC, alors qu'en vitesse il est difficile de donner une comparaison sur les performances. Ainsi l'algorithme ATRAPP (et sa variante l'algorithme TRAPP) bénéficie de l'information contenue dans les observations, au prix d'un coût de calcul plus élevé. L'algorithme TRAPP semble aussi performant que l'ATRAPP, signifiant ainsi la nécessité du rééchantillonnage au niveau des cibles. Ce rééchantillonnage devient moins nécessaire lorsque le nombre de cibles diminue, l'avantage de l'ATRAPP est donc de pouvoir s'adapter au nombre de cibles présentes (mais qui ici sont en nombre constant).

D'après les graphiques de la figure 4.6, les cibles 2 et 5 sont au plus proche au pas de temps 14 (distance d'environ 50m), et les cibles 1 et 4 sont au plus proche au pas de temps 24 (distance d'environ 30m). Ces rapprochements génèrent une augmentation de l'erreur plus importante pour l'algorithme ATRAPP (et ses variantes) que pour l'algorithme IPMCMC, ce dernier se révélant donc plus robuste au croisement des cibles. Finalement la cible 3 reste plutôt éloignée des autres cibles, l'erreur d'estimation s'améliore pour tous les algorithmes pour cette cible avec le temps, puisqu'elle se rapproche du capteur, son SNR augmente, et il devient donc plus facile pour les algorithmes de l'estimer. A l'inverse la cible 2 s'éloignant du capteur, son

erreur augmente (et ce pour tous les algorithmes) avec le temps. La cible 5 reste à peu près à une distance constante du capteur (voire se rapproche légèrement) mais son erreur augmente. La cible 5 se rapprochant du bord du champ de vision du capteur, une partie de l'information concernant cette cible n'est plus captée, et il devient alors plus difficile pour les algorithmes de continuer à l'estimer.

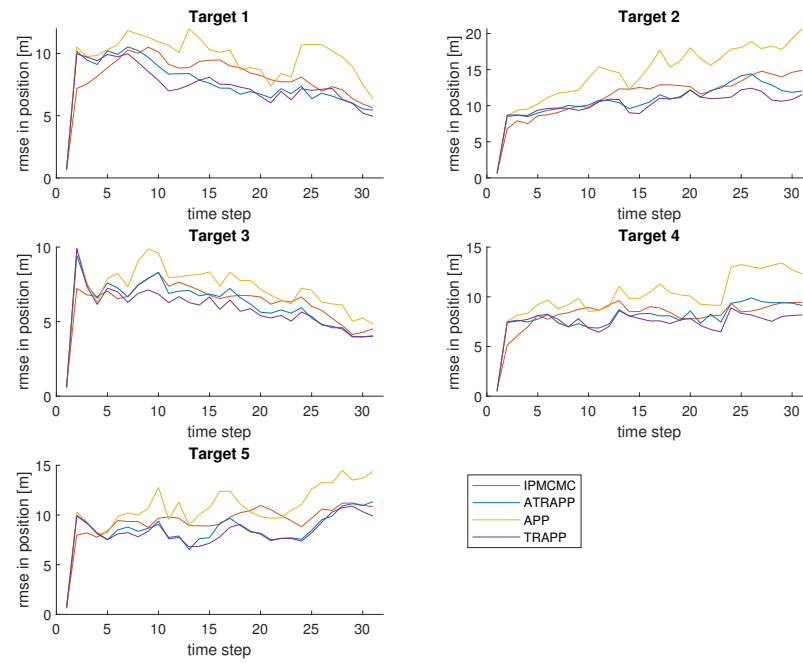


Figure 4.8: RMSE en positions

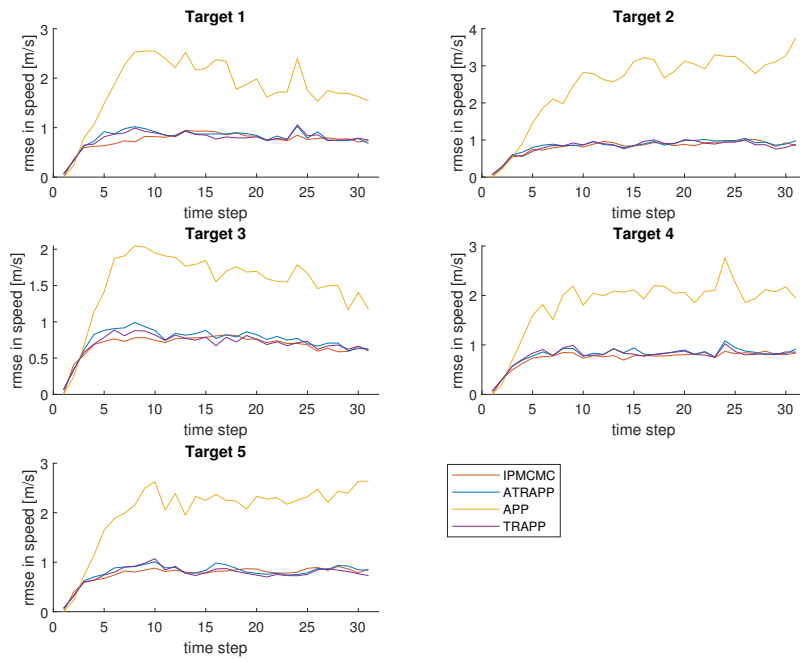


Figure 4.9: RMSE en vitesses

Chapitre 5

Choix de modélisation du système

Les méthodes et équations ont été précédemment présentées dans le cas général, une étape de détermination des modèles dynamiques et d'observation à utiliser est nécessaire. Le modèle dynamique est considéré comme étant le même pour chaque cible et est donc décrit pour une cible. Les cibles sont considérées comme des objets ponctuels en mouvement rectiligne uniforme.

5.1 Modèle dynamique

Le modèle choisi est le modèle à vitesse quasi constante, ou Nearly Constant Velocity (NCV) model.

Les cibles évoluent dans le plan cartésien, et le vecteur d'état x_t d'une cible contient les paramètres cinématiques de position et de vitesse, $(x_t \dot{x}_t y_t \dot{y}_t)$, ainsi qu'une mesure de l'amplitude du signal retour donnée par la cible I_t

$$x_t = (x_t \dot{x}_t y_t \dot{y}_t I_t)^T$$

où $[\cdot]^T$ est la notation pour la transposée.

Remarque. La présence du paramètre I_t directement dans le vecteur d'état x_t de la cible peut être discutable, car il ne s'agit pas d'un paramètre intrinsèque à la cible avec une interprétation physique directe. Cependant il s'agit d'un paramètre inconnu, qu'il est donc nécessaire d'estimer, qu'il soit intégré ou non au vecteur d'état, et sa valeur est liée aux caractéristiques de la cible considérée. Une autre approche consiste à l'estimer à part, ce qui en pratique se révèle équivalent à l'introduire directement dans le modèle dynamique.

Les objets se déplacent en ligne droite et à vitesse quasi constante, c'est-à-dire que la vitesse est constante à un bruit de modèle près, ce qui correspond à une petite accélération aléatoire

(de moyenne zéro). Pour la définition de ce bruit, la corrélation entre les coordonnées x et y est négligée. L'amplitude du signal I retour est supposée suivre une marche aléatoire. La période d'échantillonnage entre les instants t et $t + 1$ est notée Δ_t , et peut être constant ou non.

L'équation de la dynamique définie par (1.1) devient alors

$$x_t = m_t(x_{t-1}, v_t) = Fx_{t-1} + v_t$$

avec

$$F = \begin{pmatrix} F_x & \mathbf{0} & 0 \\ \mathbf{0} & F_y & 0 \\ \mathbf{0} & \mathbf{0} & 1 \end{pmatrix} \text{ et } F_x = F_y = \begin{pmatrix} 1 & \Delta_t \\ 0 & 1 \end{pmatrix}$$

où $\mathbf{0}$ est la matrice nulle de dimension 2×2 .

Le bruit d'état est considéré comme étant gaussien et définit comme suit

$$v_t \sim \mathcal{N}(0, Q)$$

avec

$$Q = \begin{pmatrix} \sigma_x^2 Q_x & \mathbf{0} & 0 \\ \mathbf{0} & \sigma_y^2 Q_y & 0 \\ \mathbf{0} & \mathbf{0} & \sigma_I^2 \Delta_t \end{pmatrix} \text{ et } Q_x = Q_y = \begin{pmatrix} \frac{\Delta_t^3}{3} & \frac{\Delta_t^2}{2} \\ \frac{\Delta_t^2}{2} & \Delta_t \end{pmatrix}$$

Le changement de vitesse latéral en x (resp. longitudinale en y), induit par le bruit d'état entre t et $t + 1$, est de l'ordre de $\sqrt{\sigma_x^2 \Delta_t}$ (resp. $\sqrt{\sigma_y^2 \Delta_t}$), ce qui permet de guider le choix du paramètre σ_x (resp. σ_y).

La densité de transition (1.2) est alors définie par

$$\begin{aligned} f_t(x_t | x_{t-1}) &= \mathcal{N}(Fx_{t-1}, Q) \\ &= \frac{1}{\sqrt{(2\pi)^{d_x} |Q|}} \exp \left(-\frac{1}{2} (x_t - Fx_{t-1})^T Q^{-1} (x_t - Fx_{t-1}) \right) \end{aligned}$$

où $|\cdot|$ correspond au calcul du déterminant.

5.2 Modèle d'observation

Les systèmes radar (radio detection and ranging), sonar (sound navigation and ranging) et lidar (light detection and ranging), peuvent être actifs ou passifs. Dans le cas des systèmes actifs, le principe consiste à émettre une onde électromagnétique (onde radio pour le radar, faisceau laser pour le lidar) ou acoustique (sonar) qui est réfléchi par la cible et réceptionnée en retour par le capteur. La localisation angulaire, ou en **azimut** (angle par rapport au nord), de la cible est déterminée par l'orientation de l'émetteur, la direction majoritaire dans laquelle est émise le faisceau, un traitement particulier en réception; on parle de **traitement angulaire**. Dans une direction donnée, le temps que met l'onde pour faire l'aller-retour, appelé aussi **retard**, permet de définir à quelle **distance** (ou rayon) se situe la cible; on parle de **traitement temporel** ou radial ou en distance. La modification de l'onde retour renseigne également sur la taille, la nature et la vitesse (radiale) de la cible. Notamment la **vitesse radiale** est déduite par le décalage en fréquence; on parle de **traitement Doppler**. Les systèmes dits passifs ne possèdent pas d'émetteurs mais restent à l'écoute de l'environnement dans le but de capter et détecter les émissions propres des cibles ou autres sources, dont les signaux sont alors inconnus ou mal connus. L'étude suivante présente un radar actif et s'appuie très largement sur les ouvrages [Sko62, LC02]; il existe de nombreuses autres références en traitement du signal radar. La cible est ici considérée comme ponctuelle, par opposition à une cible dite étendue.

5.2.1 Traitement temporel (en distance) et effet Doppler

On considère une onde envoyée dans la direction (angulaire) de la cible.

Signal émis

Le signal émis correspond à une forme d'onde choisie, sous forme d'une impulsion, notée $u(t)$, d'une durée T_p déterminée. Typiquement, $u(t)$ peut être une impulsion rectangulaire, ou une impulsion modulée linéairement en fréquence. Ces impulsions seront ensuite répétées, définissant des intervalles de répétition d'impulsion (pulse repetition interval, PRI), noté T_r . Ces intervalles peuvent avoir une valeur fixe, ou non, et définissent alors également une fréquence de répétition des impulsions (pulse repetition frequency, PRF). Ce signal est modulé par une fréquence porteuse afin d'être émis. L'écriture complexe des signaux est choisie ici. Le signal émis $s_e(t)$ s'écrit donc

$$s_e(t) = u(t)e^{i2\pi f_0 t}$$

où

- $u(t)$ représente l'enveloppe du signal, sous forme d'impulsions de durée T_p
- f_0 la fréquence de la porteuse

Propagation

Ce signal traverse l'environnement en un certain temps, avant d'être réfléchi par la cible puis être reçu par l'antenne. L'antenne possède un mode transmission durant lequel elle émet le signal, puis elle "écoute" les signaux retour en mode réception. Ces deux modes se suivent alternativement, lorsque l'antenne émet elle ne reçoit pas et perd donc les possibles échos arrivant à ce moment là, ce qui définit une zone dite aveugle à proximité de l'antenne. Ces notions sont illustrées sur la figure 5.1.

Le signal réfléchi reçu est alors, par rapport au signal émis,

- **retardé** dû au temps de propagation aller-retour,
- **modifié en fréquence** avec l'effet Doppler induit par la cible en mouvement,
- **atténué** suite à la propagation de l'onde dans l'environnement et sa réflexion sur la cible,
- **bruité** par l'environnement traversé et sa réception avec le matériel.

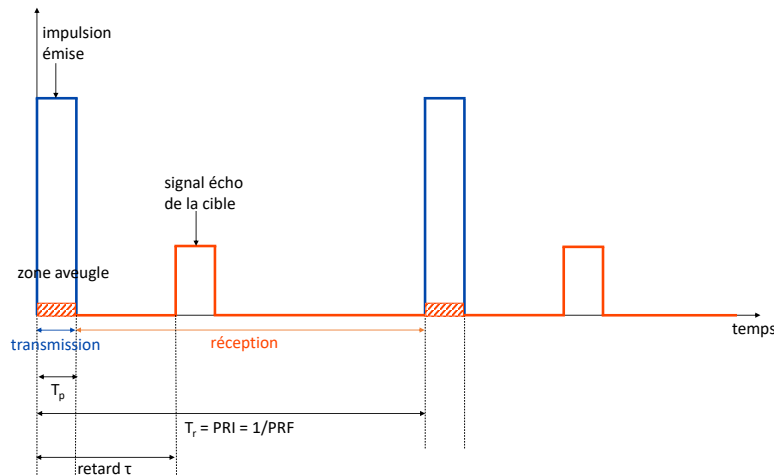


Figure 5.1: Illustration des temps du signal à l'antenne. Pour simplifier, le signal retour est ici seulement retardé et atténué.

Retard et effet Doppler

Le signal écho est reçu à l'antenne avec un retard τ . Ce retard correspond au temps qu'a mis l'onde pour faire l'aller-retour jusqu'à la cible. Connaissant la vitesse de propagation de l'onde c (vitesse de la lumière), et la distance parcourue étant alors deux fois la distance r de l'antenne à la cible, la durée de l'aller-retour, correspondant au retard, est

$$\tau = \frac{2r}{c} \quad (5.1)$$

L'expression (5.1) donne la première relation entre une information contenue dans le signal (le retard) et une donnée d'intérêt pour la position de la cible (la distance radar cible). La cible étant en mouvement, sa distance à l'antenne $r(t)$ est fonction du temps. On prendra comme origine des temps $t = 0$ l'instant de début d'émission de l'impulsion. On note t_0 (différent de $t = 0$), l'instant de référence où le signal est réfléchi sur la cible. Peuvent alors être également définis:

- r_0 la distance radar-cible au moment où l'onde radar est réfléchie sur la cible, c'est-à-dire $r_0 = r(t_0)$.
- τ_0 le retard du signal retour correspondant à r_0 , pouvant aussi être défini par $\frac{\tau_0}{2} = t_0$.

La distance et le retard de la cible définis ci-dessus sont bien sûr liés par l'expression (5.1)

$$\tau_0 = \frac{2r_0}{c} \quad (5.2)$$

La cible étant en mouvement, la distance $r(t)$ évolue lorsque l'onde est réfléchie sur la cible pendant toute la durée d'impulsion. Bien que la durée de l'impulsion T_p soit très faible par rapport à la dynamique de la cible, c'est-à-dire que la cible bouge très peu pendant cette durée, le mouvement de la cible impacte le signal. C'est d'ailleurs ce phénomène qui permet de mettre en évidence le décalage en fréquence du signal, aussi nommé effet Doppler, et permettant ainsi de retrouver la vitesse de la cible. La considération de la dépendance temporelle de $r(t)$, et donc de $\tau(t)$, dans l'expression du signal, fera apparaître l'expression de la fréquence Doppler dans ce qui suit.

On fait l'hypothèse que la vitesse radiale de la cible est constante pendant la durée de l'aller-retour de l'onde. On définit alors

$$r(t) = r_0 - v_r \times \left(t - \frac{\tau_0}{2}\right) \quad (5.3)$$

La vitesse radiale de la cible v_r est définie de telle manière qu'elle est positive lorsque la cible se rapproche du radar. C'est parfois la variation en distance $\dot{r}$ (ou *range rate*) qui est présentée,

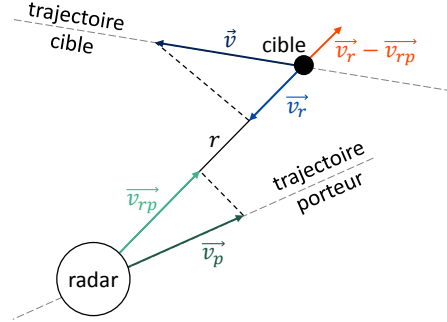


Figure 5.2: Vitesses radiales et vitesse relative résultante

et donc $v_r = -\dot{r}$. Ici il est admis que le radar est fixe et ne bouge pas. Si le porteur du radar est en mouvement, alors v_r est remplacé par la vitesse relative $v_r - v_{rp}$, avec v_{rp} la vitesse radiale du porteur, comme illustré sur la figure 5.2. Le retard (temps de l'aller-retour) est noté $\tau(t)$. Le signal est réfléchi à l'instant $t - \frac{\tau(t)}{2}$, la cible étant alors à une distance $r(t - \frac{\tau(t)}{2})$, donnant la version temporelle de l'expression (5.1) suivante

$$\tau(t) = \frac{2r(t - \frac{\tau(t)}{2})}{c} \quad (5.4)$$

et donc en injectant (5.3) dans (5.4) on obtient

$$\begin{aligned} \tau(t) &= \frac{2}{c} \left(r_0 - v_r \times \left(t - \frac{\tau(t)}{2} - \frac{\tau_0}{2} \right) \right) \\ \Rightarrow \tau(t) &= \frac{2r_0/c}{1 - v_r/c} + \frac{v_r/c}{1 - v_r/c} \tau_0 - \frac{2v_r/c}{1 - v_r/c} t \end{aligned} \quad (5.5)$$

Comme $v_r \ll c$ pour les vitesses des cibles, (5.5) peut être approché par

$$\tau(t) = \frac{2r_0}{c} + \frac{v_r}{c} \tau_0 - \frac{2v_r}{c} t \quad (5.6)$$

En utilisant (5.2) dans l'expression (5.6), on a

$$\tau(t) = \left(1 + \frac{v_r}{c} \right) \tau_0 - \frac{2v_r}{c} t \quad (5.7)$$

et à nouveau comme $v_r \ll c$, l'équation (5.7) donne finalement

$$\tau(t) = \tau_0 - \frac{2v_r}{c} t \quad (5.8)$$

Signal reçu (non bruité)

Le signal reçu $s_r(t)$ à l'instant t retardé et atténué s'écrit ¹

$$s_r(t) = \alpha u(t - \tau(t)) e^{i2\pi f_0(t - \tau(t))} \quad (5.9)$$

avec

- α un coefficient complexe, dû à l'émission, la propagation et la réflexion,

$$\alpha = \frac{1}{r^2} a e^{i\varphi} \quad (5.10)$$

$$\begin{cases} \frac{1}{r^2} & \text{atténuation} \\ a & \text{amplitude} \\ \varphi & \text{phase} \end{cases}$$

- $u(t - \tau(t))$ l'enveloppe retardée, à variation lente, qui avec l'expression du retard $\tau(t)$ de (5.8) s'écrit

$$u(t - \tau(t)) = u\left(\left(1 + \frac{2v_r}{c}\right)t - \tau_0\right)$$

Cependant la dilatation du temps apparaissant avec le facteur $(1 + \frac{2v_r}{c})$ est négligeable pour le signal $u(\cdot)$. L'impulsion ayant une durée T_p , la dilatation a un effet de l'ordre de $\frac{2v_r}{c}T_p$. Or si la bande du signal est B , alors les changements observables sont de l'ordre de $1/B$. La condition pour que l'effet de la dilatation soit négligeable, qui est respectée dans le cas des radars, s'écrit alors

$$\frac{2v_r}{c}T_p \ll \frac{1}{B}$$

ou en terme de produit du temps et de la largeur de bande de l'impulsion

$$\Rightarrow BT_p \ll \frac{c}{2v_r}$$

- $e^{i2\pi f_0(t - \tau(t))}$ la porteuse, à variation rapide, qui peut également se détailler avec (5.8)

$$\begin{aligned} e^{i2\pi f_0(t - \tau(t))} &= e^{i2\pi f_0\left(\left(1 + \frac{2v_r}{c}\right)t - \tau_0\right)} \\ &= e^{-i2\pi\tau_0} e^{i2\pi(f_0 + f_{d0})t} \end{aligned}$$

¹signal idéal encore non bruité

avec

$$f_{d0} = \frac{2f_0 v_r}{c} = \frac{2v_r}{\lambda}$$

et $\lambda = c/f_0$ longueur d'onde de la porteuse.

L'intérêt d'avoir déterminé le retard temporellement en fonction de la vitesse radiale permet ici de mettre en évidence l'effet Doppler sur la porteuse. Ainsi un glissement de fréquence est observé, définissant la fréquence Doppler f_{d0} , et qui se définit comme une rotation de phase de $e^{i2\pi f_{d0}t}$. Le terme $e^{-i2\pi\tau_0}$ est quant à lui inclus directement dans le terme de phase $e^{i\varphi}$ du coefficient complexe α défini dans (5.9) par (5.10), ainsi $\varphi \leftarrow \varphi - 2\pi\tau_0$. La phase φ étant déjà considérée comme inconnue et distribuée uniformément entre 0 et 2π , ce déphasage ne change pas ce comportement.

Le signal reçu, idéal et non bruité, lorsqu'il arrive au niveau du récepteur du radar s'exprime alors

$$s_r(t) = \alpha u(t - \tau_0) e^{i2\pi f_{d0}t} e^{i2\pi f_0 t}$$

Bruits

Trois types de bruits sont observés et affectent le signal, les deux premiers intervenant avant l'arrivée du signal au récepteur, le troisième apparaissant après la réception (et ce dernier sera donc repris au paragraphe suivant 5.2.1):

- le clutter lié à l'environnement

On appelle clutter, ou fouillis, tous les échos qui vont être renvoyés par des éléments naturels de l'environnement, et qui en s'accumulant vont générer un bruit de fond. Le clutter est souvent considéré comme immobile ou à faible vitesse par rapport à celles des cibles, il peut donc être éliminé par filtrage de la fréquence Doppler. Le clutter atmosphérique est dû à la réflexion des ondes sur les nuages. Il se définit par un paramètre de surface équivalente, qui bien souvent est inférieure à l'écho d'une cible. Il s'agit donc de réguler la détection pour s'en affranchir. A la différence du clutter atmosphérique, le clutter de sol a lui un niveau de rétrodiffusion plus élevé et supérieur à celui des cibles, et est plutôt immobile. Il peut être ainsi repéré par sa caractéristique de forte puissance, ou par traitement Doppler. Enfin le clutter de mer est un intermédiaire aux deux précédents, se comportant comme l'atmosphérique par mer calme, mais comme celui de sol par mer plus agitée (avec une vitesse faible).

- les brouilleurs

Le brouillage correspond au fait d'envoyer des signaux au radar dans le but de le tromper sur la présence et la position de cibles réelles. Il peut s'agir de fausses cibles, ou d'une émission d'un signal très puissant, supérieur à l'écho de la cible, qui sature le récepteur, dégrade le rapport signal sur bruit, rendant ainsi la détection impossible. Il faut cependant que l'énergie du brouilleur soit concentrée dans la bande passante du récepteur, mais elle ne subit l'atténuation par rapport à la distance parcourue que sur un aller, contrairement au signal qui fait l'aller-retour entre le radar et la cible.

- le bruit thermique

Le bruit thermique, noté $n(t)$, est dû au matériel interne au radar, généré dans les résistances dû au mouvement aléatoire des électrons. Il s'agit d'un bruit blanc, c'est-à-dire de densité spectrale constante. Il se modélise bien par un bruit additif gaussien complexe circulaire symétrique [Goo63] de moyenne nulle. Son autocorrélation s'écrit $R_n(\tau) = \mathbb{E}[n(t)\bar{n}(t-\tau)] = \sigma_n^2\delta(\tau)$. Circulaire symétrique signifie ici que la partie réelle et la partie imaginaire sont indépendantes et de même variance, c'est-à-dire, si $\eta = [\text{Re}(n), \text{Im}(n)]'$, alors $\mathbb{E}[\eta'\eta] = \frac{\sigma^2}{2} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$. Deux représentations de ce bruit complexe peuvent être adoptées:

- projection cartésienne

Dans le cas de la représentation cartésienne, le bruit est décomposé en partie réelle et partie imaginaire. Ces deux composantes sont deux variables gaussiennes centrées, de même variance $\frac{\sigma_n^2}{2}$ et indépendantes.

$$n(t) = \text{Re}(n) + i\text{Im}(n)$$

avec

$$\text{Re}(n) \sim \mathcal{N}(0, \frac{\sigma_n^2}{2}) \quad \text{et} \quad \text{Im}(n) \sim \mathcal{N}(0, \frac{\sigma_n^2}{2})$$

- projection polaire

Dans la représentation polaire, le bruit est défini par un module ρ , qui suit une distribution de Rayleigh, et une phase θ distribué uniformément sur $[0, 2\pi)$.

$$n(t) = \rho e^{i\theta}$$

avec

$$\rho \sim \text{Rayleigh}(\sigma) \quad \text{et} \quad \theta \sim \mathcal{U}([0, 2\pi))$$

et on vérifie $\mathbb{E}[\rho^2] = \mathbb{E}[n\bar{n}] = \sigma_n^2$.

Le clutter de mer ainsi que les brouilleurs, s'ajoutant au signal pendant la propagation, ne seront pas développés dans la suite. Le bruit thermique s'ajoutant juste après réception du signal, sera quant à lui pris en compte dans ce qui suit.

Réception et chaîne de traitement du signal

L'onde arrive au niveau du récepteur du radar et entre dans la chaîne de traitement du signal [NS97].

Amplificateur Le signal est amplifié. L'amplitude du signal est modifiée, donc a défini en (5.10) est désormais l'amplitude amplifiée, qui sera appelée simplement amplitude dans la suite. Au signal utile s'ajoute alors un bruit blanc thermique $n(t)$. A cette étape, le signal bruité est

$$s_r(t) = \alpha \underbrace{u(t - \tau_0)e^{i2\pi f_{d0}t}}_{s(t, \tau_0, f_{d0})} e^{i2\pi f_0 t} + n(t) \quad (5.11)$$

avec $s(t, \tau_0, f_{d0})$ le signal contenant les informations d'intérêt sur la cible. Ainsi les signaux sont composés de signaux utiles (écho de cibles) mélangés à des signaux parasites. Le rôle du traitement du signal est:

- d'éliminer au mieux les signaux parasites (bruit)
- de détecter les cibles
- d'estimer les paramètres des cibles

Mélangeur Le mélangeur couplé à un oscillateur local permettent de convertir la fréquence porteuse en fréquence intermédiaire f_i . Ce procédé permet en pratique de faciliter l'étape qui suit, mais n'a aucun impacte sur l'étude théorique suivante. Pour cette raison, on omettra désormais la composante relative à la fréquence porteuse ou fréquence intermédiaire.

Filtre adapté L'enjeu désormais va être de faire le traitement nécessaire pour extraire du signal reçu décrit par (5.11) les informations d'intérêt que sont le retard τ_0 et la fréquence Doppler f_{d0} , donnant respectivement la distance et la vitesse radiale de la cible, appelés paramètres et formant le couple (τ_0, f_{d0}) . Il s'agit du problème dit d'estimation. En l'absence de la cible, le signal émis n'est jamais retourné au radar. Le problème de détection est ainsi soulevé, c'est-à-dire déterminer s'il y a une cible ou non, dans un premier temps, afin ensuite de pouvoir en extraire les paramètres si une cible existe. L'estimation est donc une recherche de paramètres tandis que la détection est une décision entre présence ou absence de la cible. Cette décision est prise à l'aide d'un seuil sur la valeur du signal traité par le récepteur; lorsque le seuil est dépassé, la cible est déclarée. Cette réflexion mène à la détermination d'un mode opératoire pour le traitement du signal reçu. Les radars traitent le signal entrant de manière à ce que la réception soit optimale selon un critère défini. Pour la détection, le critère est celui de Neyman-Pearson, c'est-à-dire que la probabilité de détection de la cible doit être la plus élevée possible en assurant une probabilité de fausse alarme constante et prédéterminée. Une fausse alarme apparaît lorsqu'une cible est déclarée alors qu'il n'y en a pas, ce qui n'est pas souhaitable, on cherche donc à ce que la probabilité de fausse alarme soit faible. Mais dans le cas de la problématique de pistage avant détection, justement la détection n'est pas requise. En revanche, on s'intéresse à l'estimation des paramètres de la cible. Pour cette tâche, le critère du maximum de vraisemblance est celui retenu. C'est-à-dire que pour des jeux de paramètres possibles (τ, f_d) , on cherche celui qui conduirait le plus probablement à observer le signal effectivement reçu et bruité. Les deux critères énoncés ci-dessus mènent finalement à la même définition du récepteur optimal, donc la même chaîne de traitement du signal. Seule la dernière étape en bout de chaîne est différente, le détecteur décide par seuil de l'existence ou non d'une cible en chaque point, l'estimateur donnant quant à lui le jeu de paramètre qu'il estime le plus probable.

Ce récepteur optimal tel que décrit ci-dessus prend alors la forme d'un corrélateur entre le signal reçu $s_r(t)$ avec des répliques du signal attendu (non bruité) pour des jeux de paramètres donnés, c'est-à-dire autant de jeux que de positions possibles de l'écho issu de la cible, qui forment alors les points de l'image radar. On parle également de **filtre adapté**. Ce récepteur découle de l'hypothèse d'un bruit additif blanc gaussien $n(t)$ et n'est donc idéal que dans ce cas. De ce filtrage sort un nouveau signal transformé par la corrélation, débruité au mieux, contenant toujours les informations d'intérêt de la cible qu'il sera plus facile d'extraire. Cette nouvelle forme fait apparaître la **fonction d'ambiguïté**. Elle dépend des paramètres de la cible, mais surtout de la forme d'onde initialement émise, c'est-à-dire $u(t)$. La fonction d'ambiguïté est manipulable via le choix de la forme de l'onde transmise, pour avoir de bonnes propriétés permettant une estimation des paramètres fiable, précise, avec une bonne résolution.

Finalement, la sortie du filtrage adapté $s_{r,FA}(t)$ est donnée le filtrage adapté donne la fonction d'ambiguïté sous la forme suivante

$$\begin{aligned} s_{r,(\tau,f_d)} &= \int s_r(t) \bar{s}(t, \tau, f_d) dt \\ &= \alpha \underbrace{\int s(t, \tau_0, f_{d0}) \bar{s}(t, \tau, f_d) dt}_{\chi(\tau, \tau_0, f_d, f_{d0})} + \int n(t) \bar{s}(t, \tau, f_d) dt \end{aligned}$$

avec $\chi(\tau, \tau_0, f_d, f_{d0})$ la fonction d'ambiguïté, déterminée par le choix de la forme d'impulsion $u(t)$.

Le but du traitement du radar est donc de traiter de façon optimale le signal pour s'affranchir au mieux du bruit et résoudre les problèmes de détection et d'estimation. Le bruit considéré est le bruit additif $n(t)$. Pourtant il n'est pas la seule donnée de l'équation à générer un caractère aléatoire et donc à rendre plus difficilement prévisible le signal retour. Le coefficient complexe α défini par (5.10) est également inconnu, composé de la phase et de l'amplitude, tous les deux aléatoires.

5.2.2 Traitement angulaire

Le traitement temporel précédemment présenté de l'onde suppose que le radar émet en direction de la cible et reçoit le signal retour provenant de cette même direction. Cette direction correspond à la localisation angulaire de la cible, en azimuth et élévation, que l'on souhaite également estimer. On ne traitera cependant que l'azimut dans ce qui suit. L'obtention d'un signal dirigé finement dans la direction voulue n'est pas immédiat, un élément d'antenne isotrope rayonnant uniformément dans toutes les directions par définition. Il en est de même en réception pour la sélection angulaire et définir la provenance du signal reçue. Pour contrôler la directivité de l'antenne, dont dépend la qualité de l'estimation de l'azimut de la cible, des techniques de **beamforming**, aussi appelé formation de faisceau, formation de voie ou encore filtrage spatial, sont utilisées. Ceci est possible techniquement à l'aide de réseaux d'antennes, en combinant les signaux des éléments d'antennes isotropes, afin qu'ils forment des interférences constructives dans certaines directions (dont une direction majoritaire, celle qui nous intéresse, dans laquelle les signaux s'ajoutent) et des interférences destructives dans les autres directions (dans lesquelles les signaux s'atténuent voire s'annulent). Selon l'assemblage des éléments du réseau, c'est-à-dire leur nombre et leurs dispositions, un diagramme de rayonnement peut être défini. La direction majoritaire vers laquelle le faisceau pointe en rayonnant le plus d'énergie

définit généralement le lobe principal. Mais il existe aussi des directions (indésirables) vers lesquelles du signal, plus faible, est tout de même dirigé, formant les lobes secondaires. Le beamforming peut s'opérer en émission et en réception. En émission il permet d'orienter le faisceau dans une direction. En réception il permet de sélectionner un faisceau de signal reçu en mettant les signaux en phase selon la direction souhaitée, il s'agit d'un filtrage spatial du signal qui arrive. On traite ici le beamforming en réception, mais par réciprocity les résultats sont équivalents en émission.

Afin de couvrir toutes les directions, le balayage du faisceau peut être mécanique ou électronique. Dans le cas du balayage mécanique, l'antenne réseau est dite fixe, le faisceau pointe majoritairement dans une direction (le diagramme de rayonnement est fixe). Alors l'antenne est physiquement déplacée et tourne pour couvrir toutes les directions dans lesquelles on souhaite émettre et recevoir de l'information. Dans le cas du balayage électronique, une antenne réseau à commande de phase permet de retarder (décaler en phases) individuellement chacun des signaux des éléments de l'antenne afin d'orienter les interférences constructives dans la direction d'intérêt, et ainsi de balayer au choix les directions. Le décalage de phase des signaux et leur sommation cohérente peut être analogique ou digitale.

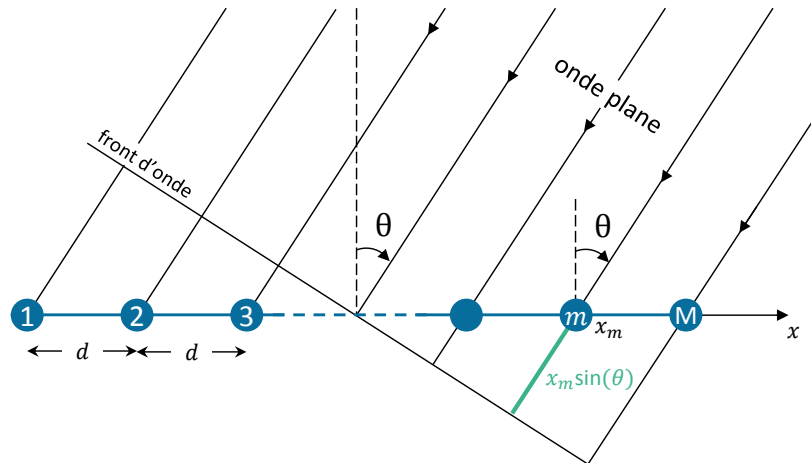


Figure 5.3: Antenne réseau à commande de phase uniforme linéaire

Antenne réseau et déphasage

On considère une antenne réseau composée de M éléments isotropes espacés uniformément d'une distance d sur une ligne, comme sur la figure 5.3, communément appelée un réseau linéaire uniforme (uniform linear array, ULA). Les éléments étant uniformément espacés, le déphasage est le même entre chaque élément. Les éléments de l'antenne sont positionnés aux abscisses:

$$x_m = \left(m - \frac{M+1}{2}\right) d, \quad m = 1, \dots, M$$

Le signal, représenté par une onde plane selon l'approximation champs lointain (la cible est suffisamment éloignée), arrive sur le réseau avec une incidence θ (voir figure 5.3). En considérant comme origine des phases le centre du réseau d'antennes, le déphasage $\Delta\varphi_m(\theta)$ du signal à l'élément d'antenne m se définit alors

$$\Delta\varphi_m(\theta) = \omega\tau_m(\theta)$$

avec

- ω la pulsation de l'onde ($\omega = 2\pi f$)
- $\tau_m(\theta)$ le temps nécessaire à l'onde pour parcourir la distance $x_m \sin(\theta)$, soit $\tau_m(\theta) = \frac{x_m}{c} \sin(\theta)$.

Ainsi le déphasage du signal à l'élément m est:

$$\Delta\varphi_m(\theta) = \frac{2\pi}{\lambda} x_m \sin(\theta) \tag{5.12}$$

avec λ la longueur d'onde du signal ($\lambda = \frac{c}{f}$).

De la même manière, le déphasage "élémentaire" entre chaque élément séparés par une distance d est déterminé par $\Delta\varphi(\theta) = \frac{2\pi}{\lambda} d \sin(\theta)$, et on peut alors aussi exprimer $\Delta\varphi_m(\theta)$ de l'équation (5.12) en fonction de $\Delta\varphi(\theta)$ de la manière suivante

$$\Delta\varphi_m(\theta) = \left(m - \frac{M+1}{2}\right) \Delta\varphi(\theta)$$

Principe du beamforming

Soit un signal $s(t)$ provenant de la direction θ , comme sur la figure 5.4, et arrivant sur l'antenne. Sa composante arrivant sur l'élément m est déphasée de $\Delta\varphi_m(\theta)$ et s'écrit donc

$$s_m(t) = s(t)e^{j\Delta\varphi_m(\theta)}$$

On peut définir un vecteur de direction (steering vector) $\mathbf{w}(\theta)$, lié à l'angle physique θ et définit par:

$$\mathbf{w}(\theta) = [e^{j\Delta\varphi_1(\theta)}, \dots, e^{j\Delta\varphi_M(\theta)}]^T \quad (5.13)$$

et le vecteur $\mathbf{s}(t) = [s_1(t), \dots, s_M(t)]^T$ des signaux reçus aux éléments s'écrit alors

$$\mathbf{s}(t) = \mathbf{w}(\theta)s(t)$$

En sommant les signaux reçus à chacun des éléments (sans les remettre en phase) afin de voir comment les interférences agissent, et en considérant l'amplitude du signal unitaire et constante, on génère le diagramme de rayonnement de l'antenne fixe (normalisé par $1/M$):

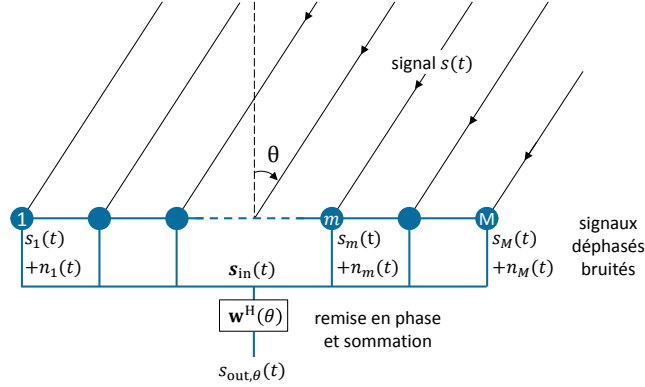
$$\begin{aligned} D(\theta) &= \frac{1}{M} \sum_{m=1}^M e^{j(m - \frac{M+1}{2})\Delta\varphi(\theta)} \\ &= \frac{1}{M} \frac{\sin\left(\frac{M\Delta\varphi(\theta)}{2}\right)}{\sin\left(\frac{\Delta\varphi(\theta)}{2}\right)} \end{aligned}$$

Le diagramme de rayonnement est représenté figure 5.6 en bleu, avec une antenne composée de 10 éléments, espacés d'une distance $d = \frac{\lambda}{2}$. Ainsi, le lobe principal du radar pointe dans la direction perpendiculaire à l'axe de l'antenne. On souhaite définir un traitement afin d'orienter le faisceau selon une direction souhaitée.

Un bruit blanc complexe circulaire gaussien s'ajoute sur chacun des signaux des éléments, et en définissant le vecteur des bruits $\mathbf{n}(t) = [n_1(t), \dots, n_M(t)]^T$, le signal bruité (vecteur des signaux) en réception s'écrit

$$\mathbf{s}_{\text{in}}(t) = \mathbf{s}(t) + \mathbf{n}(t)$$

Pour filtrer la direction θ d'intérêt, le but est de maximiser le rapport signal sur bruit et de minimiser la puissance du bruit en sortie dans cette direction. Pour ce faire la solution optimale

Figure 5.4: Beamforming pour la direction θ

du beamforming est de remettre en phase les signaux reçus à chaque élément d'antenne selon la direction que l'on souhaite filtrer et les sommer (de façon cohérente car en phase). Cela revient à donc à appliquer

$$s_{\text{out},\theta}(t) = \frac{\mathbf{w}(\theta)^H}{\mathbf{w}(\theta)^H \mathbf{w}(\theta)} s_{\text{in}}(t)$$

avec $(\cdot)^H$ le conjugué transposé (opérateur hermitien). Le conjugué transposé du vecteur direction, $\mathbf{w}^H(\theta)$, permet donc de remettre en phase les signaux et de les sommer. Le dénominateur $\mathbf{w}(\theta)^H \mathbf{w}(\theta) = M$ (en utilisant l'équation (5.13)) permet de normaliser. Chaque élément du vecteur direction à un module unitaire, mais il est possible de les pondérer pour modifier le diagramme de rayonnement, comme par exemple en appliquant un fenêtrage de Hamming. La pondération de chaque terme permet de contrôler l'amplitude des lobes secondaires et la largeur du lobe principal, tandis que les phases contrôlent la direction du lobe principal (centré en θ) et les nuls (les directions dans lesquelles le signal s'annule). Le beamforming peut être vu comme un filtre adapté spatial, sur les angles d'arrivée.

Beamforming et balayage

La cible est située à θ_0 (voir figure 5.5). Le signal réfléchi sur la cible contribue donc au signal lorsque le faisceau est filtré dans sa direction, mais aussi lorsqu'elle intervient dans les lobes secondaires sur d'autres directions. Le vecteur des signaux reçus aux éléments de l'antenne et bruité s'écrit donc

$$\mathbf{s}_{r,\text{in}}(t) = \mathbf{w}(\theta_0) s_r(t) + \mathbf{n}(t)$$

Le radar va balayer les différentes directions à observer. Dans le cas d'un radar à balayage mécanique, l'antenne réseau permet d'orienter le faisceau dans une direction et filtre également en réception dans cette direction. Ainsi toutes les directions angulaires seront couvertes successivement. Dans le cas de l'antenne réseau à commande de phase, en réception, si le beamforming est analogique, alors une seule direction peut être filtrée à la fois. Si le beamforming est digital, il est possible de le réaliser pour plusieurs directions en même temps. Dans ce dernier cas, plusieurs angles sont filtrés, les bruits sur chaque angle sont cependant corrélés. Généralement en radar, en émission l'énergie est concentrée dans une direction (beamforming en émission), et donc en réception il est préférable de ne filtrer que dans cette direction [LC02]. La figure 5.5 illustre donc les différentes directions filtrées. La sortie du beamforming après filtrage selon θ donne

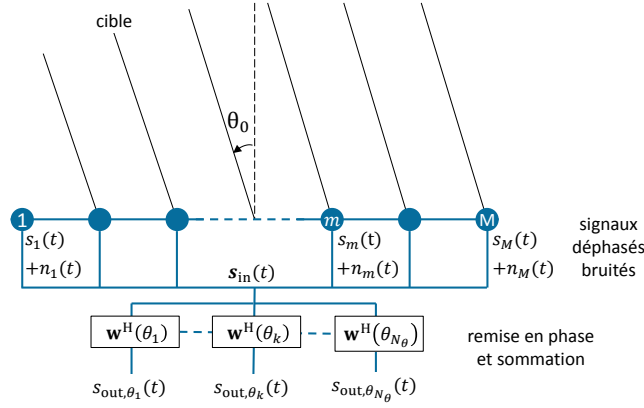


Figure 5.5: Beamforming selon plusieurs directions (simultanément ou successivement)

$$\begin{aligned}
 s_{r,\text{out},\theta}(t) &= \frac{1}{M} \mathbf{w}^H(\theta) \mathbf{s}_{r,\text{in}}(t) \\
 &= \underbrace{\frac{1}{M} \mathbf{w}^H(\theta) \mathbf{w}(\theta_0)}_{D(\theta, \theta_0)} s_r(t) + \underbrace{\frac{1}{M} \mathbf{w}^H(\theta) \mathbf{n}(t)}_{n_\theta(t)}
 \end{aligned} \tag{5.14}$$

Ainsi au signal reçu $s_r(t)$ est appliqué un diagramme défini par

$$D(\theta, \theta_0) = \frac{1}{M} \frac{\sin\left(M \frac{\pi d}{\lambda} (\sin(\theta_0) - \sin(\theta))\right)}{\sin\left(\frac{\pi d}{\lambda} (\sin(\theta_0) - \sin(\theta))\right)}$$

Le diagramme $|D(\theta, \theta_0)|$ est représenté figure 5.6. Lorsque $\theta_0 = 0$, on retrouve le di-

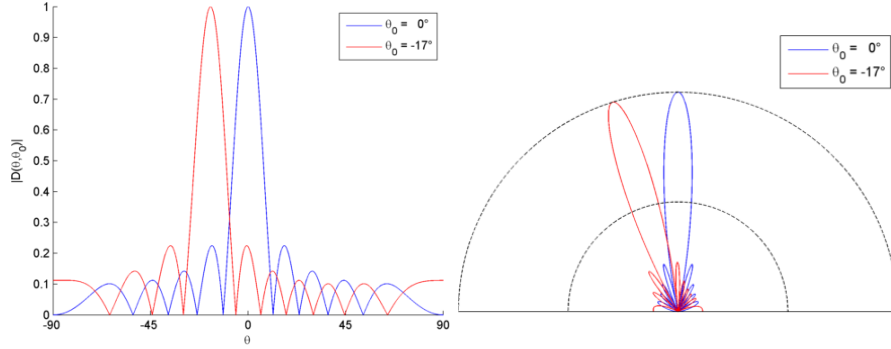


Figure 5.6: Diagrammes de rayonnement $|D(\theta, \theta_0)|$, $M = 10$, $d = \frac{\lambda}{2}$.

agramme de rayonnement fixe de l'antenne. Le bruit reste gaussien complexe, et est noté $n_\theta(t) = \frac{1}{M} \mathbf{w}^H(\theta) \mathbf{n}(t)$. L'équation (5.14) du signal en sortie de beamforming s'écrit donc

$$s_{r,\text{out},\theta}(t) = D(\theta, \theta_0) s_r(t) + n_\theta(t)$$

Concrètement, l'étape du beamforming intervient avant le filtrage adapté, mais le traitement en distance et le traitement angulaire étant découplés, ils affectent indépendamment l'un de l'autre le signal reçu.

5.3 Calcul de vraisemblance

Afin d'évaluer l'adéquation entre les connaissances de l'état du système, c'est-à-dire ce qui est estimé, et les observations réellement faites, la fonction de vraisemblance $p(z|x)$ doit être définie. Elle définit la probabilité que la mesure réelle soit effectivement observée compte tenu de l'estimation de l'état des cibles. Comme vu précédemment, c'est une composante des équations de filtrage. Cependant la vraisemblance n'est pas directement accessible, puisque la relation entre les observations et l'état des cibles dépend également des paramètres du coefficient complexe: l'amplitude a et la phase φ , inconnus et aléatoires. A un instant donné et pour une cible donnée, la valeur de ces paramètres inconnus est la même dans toutes les cellules de résolution de l'image radar, contrairement à la fonction d'ambiguïté qui elle varie en chaque pixel. C'est en cela que consiste le principe de cohérence spatiale. Un usage judicieux de cette puissante caractéristique permet ainsi de conserver de l'information. Les termes d'amplitude d'une cible et phase d'une cible seront utilisées pour parler de l'amplitude et de la phase du coefficient complexe du signal renvoyé par la cible de manière plus succincte.

5.3.1 Rappel du modèle d'observation

L'image radar s'exprime selon les états des cibles, leurs amplitudes et leur phases, formant ainsi la fonction d'observation, de la manière suivante:

$$z = \sum_{j=1}^N a_j e^{i\varphi_j} f(x_j) + n \quad (5.15)$$

avec

- $z \in \mathbb{C}^{N_c}$ l'image complexe radar (N_c le nombre de cellules)
- $a_j \in \mathbb{R}^+$ l'amplitude de la cible j
- $\varphi_j \in [0, 2\pi)$ la phase de la cible j
- $f(\cdot) \in \mathbb{R}^{N_c}$ la fonction d'équipement radar
- $x_j \in \mathcal{X}$ le vecteur d'état de la cible j ($\mathcal{X}$ l'espace d'état)
- $n \in \mathbb{C}^{N_c}$ le bruit blanc additif gaussien complexe circulaire symétrique.

Le bruit n a une moyenne nulle et une matrice de covariance $\Sigma_n \in \mathbb{R}^{N_c \times N_c}$. Le bruit a une distribution gaussienne complexe circulaire symétrique se notant alors $\mathcal{CN}(0, \Sigma_n)$.

On note

$$\mu = \mu(x, a, \varphi) = \sum_{j=1}^N a_j e^{i\varphi_j} f(x_j) \quad (5.16)$$

et l'équation d'observation (5.15) s'écrit

$$z = \mu + n \quad (5.17)$$

5.3.2 Modèles des paramètres phase et amplitude

Phase φ

La phase varie au cours du temps. De plus il n'y a pas de cohérence dynamique, ou de façon équivalente temporelle, pour la phase du signal reçu de la cible. Aucune information ne peut être récupérée d'un traitement temporel de cette variable. Pour une cible donnée et à chaque instant, la phase est distribuée uniformément sur l'intervalle $[0, 2\pi)$.

$$\varphi \sim \mathcal{U}([0, 2\pi))$$

La phase générée par la cible j est notée φ_j .

Amplitude a

Définition générale En pratique, l'amplitude du signal retour peut ne pas être constante, et changer dans le temps, selon l'orientation de la cible, ou en raison de facteurs externes, liés à l'environnement par exemple. L'étude de l'amplitude dépend du modèle considéré pour son comportement. Ces modèles sont connus sous le nom de modèles de Swerling. Dans le cas du modèle de fluctuation Swerling 0 de l'amplitude, cette dernière est considérée comme constante dans le temps. Dans le cas des modèles de fluctuation Swerling 1, 2, 3 ou 4 de l'amplitude, cette dernière n'est pas constante. De plus il n'y a pas de cohérence dynamique, ou de façon équivalente temporelle, de l'amplitude de la cible. Ce qui est cohérent temporellement, c'est la moyenne de cette amplitude. Cette moyenne est considérée comme une constante, et aidera donc à la définition de la densité de probabilité selon laquelle l'amplitude est distribuée.

Les modèles de fluctuation de Swerling [Swe60, Sko62] définissent initialement le comportement de la surface équivalente radar (SER) de la cible, notée σ , qui est caractéristique d'une cible et intervient dans l'équation radar et donc dans la forme du signal retour observé en provenance de la cible. Le fluctuation de cette SER est soit considérée comme constante, soit fluctuante suivant le modèle générique suivant

$$p(\sigma) = \frac{m^m}{\Gamma(m)\sigma_{av}^m} \sigma^{m-1} e^{-\frac{m\sigma}{\sigma_{av}}} \quad \sigma \geq 0$$

avec $m \in \{1, 2\}$, $\Gamma(\cdot)$ la fonction gamma, $\mathbb{E}[\sigma] = \sigma_{av}$ la moyenne, et $\text{Var}(\sigma) = \frac{\sigma_{av}^2}{m}$ la variance. On peut noter que pour $m = 1$ et $m = 2$ on a $\Gamma(m) = 1$. La moyenne ne dépend pas de m , tandis que la variance est inversement proportionnelle à m ; lorsque m augmente, la fluctuation diminue². Cette distribution correspond à une distribution du khi-deux avec 2 degrés de liberté ($m = 1$) ou 4 degrés de liberté ($m = 2$), mise à l'échelle³. On remarque que la variable $\frac{m\sigma}{\sigma_{av}}$ suit une loi gamma $\Gamma(m, 1)$ c'est-à-dire que la variable $2\frac{m\sigma}{\sigma_{av}}$ suit une loi du khi-deux à $2m$ degrés de liberté. En fait, l'amplitude du signal se comporte comme la racine carrée de la surface équivalent radar, à un facteur multiplicatif près. Les modèles de Swerling définissent donc la densité de probabilité de l'amplitude au carré, à partir de laquelle la distribution de l'amplitude peut alors être déduite.

L'amplitude générée par la cible j est notée a_j , et le paramètre de sa distribution correspondante γ_{a_j} .

²lorsque m est infini, il n'y a plus de fluctuation, c'est le cas constant.

³mise à l'échelle signifie que si $x \sim p_x(x)$ alors $y = Cx \sim \frac{1}{C}p_x(\frac{y}{C})$, avec C un coefficient multiplicatif. Par exemple ici c'est $x = \frac{2m\sigma}{\sigma_{av}} \sim p_x(x)$ avec $p_x(\cdot)$ la densité classique du khi-deux dont on connaît l'expression explicite. Et donc on a $\sigma = Cx \sim \frac{1}{C}p_x(\frac{\sigma}{C})$ avec $C = \frac{\sigma_{av}}{2m}$.

Modèles de Swerling (pour l'amplitude)

Swerling 0 L'amplitude a de chaque cible est supposée constante et égale à l'inconnue γ_a . La densité de probabilité correspondante est alors

$$p(a) = \delta(a - \gamma_a) \quad (5.18)$$

où $\delta(\cdot)$ est la fonction delta de Dirac. De cette manière, l'amplitude perd son caractère aléatoire. L'amplitude est simplement le paramètre statique inconnu à estimer, égal à γ_a . Sa valeur estimée est ainsi directement incluse dans les équations.

Fluctuation inter-balayage ou inter-impulsion Les modèles de Swerling définissent la forme de la densité de probabilité à partir de laquelle l'amplitude est extraite, mais aussi la rapidité de la fluctuation de l'amplitude. Concernant cette dernière caractéristique, deux manières sont considérées pour la fluctuation de l'amplitude: inter-balayage ou inter-impulsion. Cela ne concerne pas le modèle de Swerling 0 puisque la valeur est constante, comme expliqué ci-avant, et donc ne fluctue pas.

Un balayage est une période (de rotation) du radar, c'est-à-dire le temps durant lequel il couvre tout son champ de vision, illuminant à un moment la cible lorsque le faisceau passe sur celle-ci. Une impulsion est une émission courte du radar qui est réfléchiée par la cible. Les impulsions sont émises à intervalles définis lors du balayage. Le temps d'illumination est le temps durant lequel la cible est illuminée par le faisceau du radar. Durant ce temps, plusieurs impulsions radar atteignent la cible et sont réfléchies par celle-ci.

D'après ces considérations et définitions sur les différents temps, deux comportements pour le paramètre d'amplitude sont observés. Si lors d'un balayage, ou plus précisément pendant le temps d'illumination, l'amplitude reste inchangée pour tous les signaux échos des impulsions envoyées durant ce temps, il s'agit d'une fluctuation inter-balayage. Cependant la valeur de l'amplitude change d'un balayage à un autre. C'est une variation dite lente. Au contraire, si l'amplitude prend une valeur différente pour chaque signal écho d'une impulsion, il s'agit d'une fluctuation inter-impulsion, qui varie rapidement. Ce peut être le cas dans des environnements complexes et agités par exemple.

En résumé

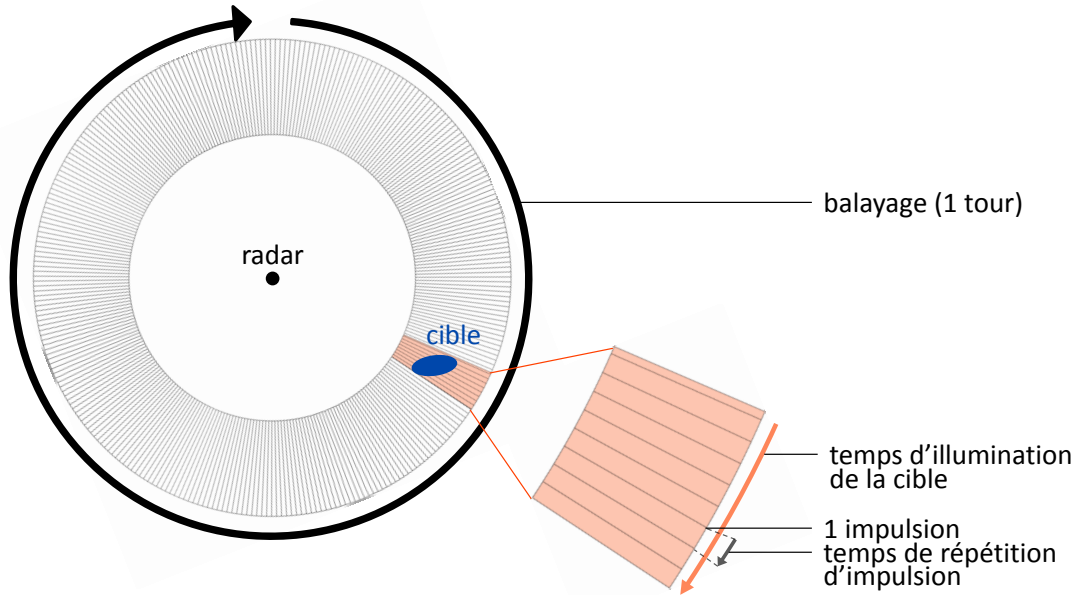


Figure 5.7: Définition des temps radar

- inter-balayage : l'amplitude est constante durant un même balayage, mais varie indépendamment d'un balayage à l'autre.
- inter-impulsion : l'amplitude varie indépendamment d'une impulsion à une autre, donc varie durant un même balayage.

Swerling 1 ou 2 Les deux modèles Swerling 1 et 2 définissent le comportement de l'amplitude avec la même densité de probabilité. La différence réside dans la rapidité de fluctuation de l'amplitude. Le modèle de Swerling 1 est inter-balayage, tandis que le modèle Swerling 2 est inter-impulsion.

L'amplitude au carré suit une distribution du khi-deux à deux degrés de liberté (mise à l'échelle), ou de manière équivalente une distribution exponentielle paramétrée par sa moyenne notée γ_a . C'est-à-dire que la variable $Y = a^2$ a pour densité

$$p_Y(y) = \frac{1}{\gamma_a} \exp\left(-\frac{y}{\gamma_a}\right)$$

avec

$$\mathbb{E}[Y] = \gamma_a$$

A partir de ce constat, l'amplitude suit une distribution du khi avec deux degrés de liberté (mise à l'échelle), ou de manière équivalente une distribution de Rayleigh. En effet, la variable a a pour densité

$$p(a) = 2ap_Y(a^2) = \frac{2a}{\gamma_a} \exp\left(-\frac{a^2}{\gamma_a}\right) \quad (5.19)$$

et ainsi

$$\mathbb{E}[a] = \frac{\sqrt{\gamma_a \pi}}{2}$$

Ces modèles sont caractéristiques d'une cible présentant des réflecteurs, ou des zones de réflexion, répartis sur une surface et à peu près de taille égale. Le cas Swerling 1 est une bonne approximation pour les objets en aviation.

Swerling 3 ou 4 Les deux modèles Swerling 3 et 4 définissent le comportement de l'amplitude avec la même densité de probabilité. La différence réside dans la rapidité de fluctuation de l'amplitude. Le modèle de Swerling 3 est inter-balayage, tandis que le modèle Swerling 4 est inter-impulsion.

L'amplitude au carré suit une distribution du khi-deux à quatre degrés de liberté (mise à l'échelle), paramétrée par sa moyenne γ_a . C'est-à-dire que la variable $Y = a^2$ a pour densité

$$p_Y(y) = \frac{4y}{\gamma_a^2} \exp\left(-\frac{2y}{\gamma_a}\right)$$

avec

$$\mathbb{E}[Y] = \gamma_a$$

A partir de ce constat, l'amplitude suit une distribution du khi avec quatre degrés de liberté (mise à l'échelle). En effet, la variable a a pour densité

$$p(a) = 2ap_Y(a^2) = \frac{8a^3}{\gamma_a^2} \exp\left(-\frac{2a^2}{\gamma_a}\right) \quad (5.20)$$

et ainsi

$$\mathbb{E}[a] = \frac{3}{4}\sqrt{\gamma_a \pi}$$

Ces modèles sont caractéristiques d'une cible composée d'une large surface de réflexion prépondérante sur plusieurs autres plus petites surfaces.

Densité de probabilité	Fluctuation	
	inter-balayage	inter-impulsion
éq. (5.19)	Swerling 1	Swerling 2
éq. (5.20)	Swerling 3	Swerling 4

Table 5.1: Récapitulatif des modèles de Swerling

Les modèles de Swerling sont résumés dans la table 5.1, et illustrés sur la figure 5.8 représentant l'amplitude d'une cible sur trois balayages complet et le début d'un quatrième, les zones "pleines" (ou grises) étant le moment d'illumination de la cible, donc les amplitudes du signal écho qui sera enregistré à ce moment. Pour les modèles de Swerling 1 et 3, la fluctuation est donc plus lente, et lors d'un temps d'illumination l'amplitude sera donc considérée comme constante. Tandis que pour les modèles de Swerling 2 et 4, la fluctuation est rapide et indépendante d'impulsion en impulsion.

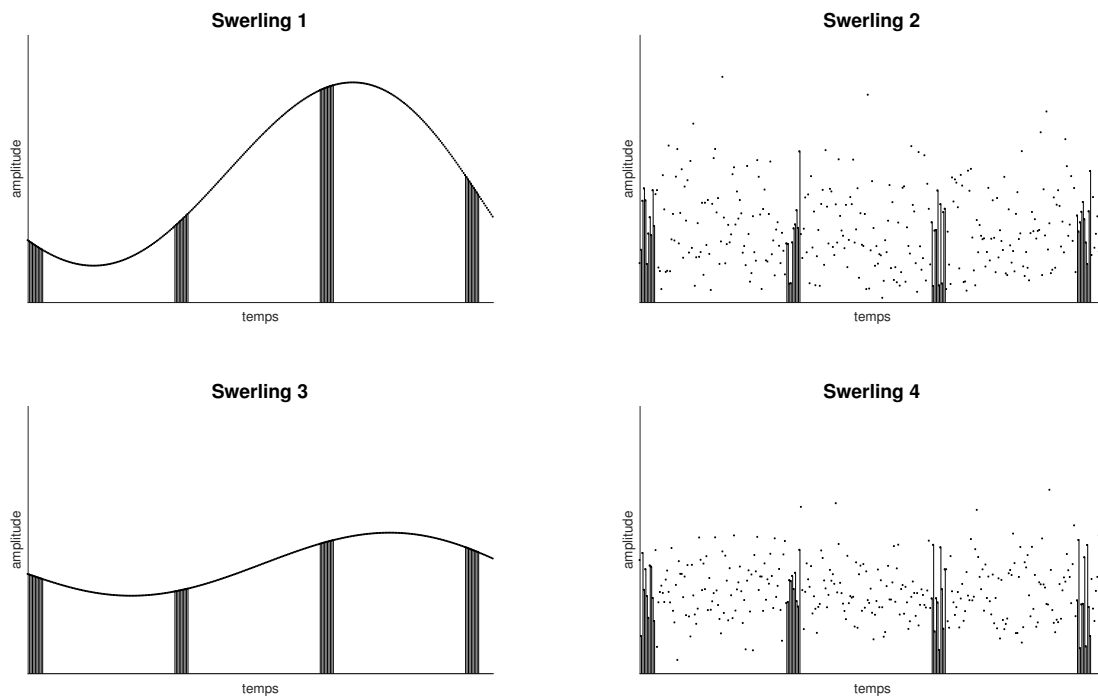


Figure 5.8: Illustration des modèles de Swerling. L'amplitude varie selon le temps, les bâtons noirs représentent les amplitudes pendant la période d'illumination de la cible par le radar.

Remarque sur le paramètre γ_a

Le paramètre γ_{a_j} , permettant de décrire la distribution de l'amplitude de la cible j , est considéré comme inconnu et est à estimer. L'espace d'état peut être augmenté afin d'inclure le paramètre dans chacun des états des cibles, ainsi l'état x_j initiale de la cible j peut être augmenté: $[x_j, \gamma_{a_j}]$. On continuera cependant dans la suite à noter cet état augmenté simplement x_j .

Bien que constants dans le temps, ces paramètres seront munis d'une légère dynamique artificielle, afin de permettre, à partir des valeurs d'initialisation qu'on leur donnera respectivement, que les valeurs estimées convergent vers leurs valeurs réelles respectives.

5.3.3 Définition et expressions des vraisemblances**Notations**

Les notations présentées ici sont adoptées dans la suite afin d'écrire des expressions de façon compacte et ainsi faciliter la lecture.

Dans le **cas multicible**, on notera x , a et φ les variables regroupant respectivement tous les états, les amplitudes et les phases des N cibles présentes, et individuellement x_j , a_j et φ_j ceux de la cible j . Une intégrale sur la variable a ou sur la variable φ correspond donc à une intégrale multiple sur tous les éléments a_j et φ_j . Pour marginaliser, on intègre ces variables sur leurs domaines de définition, c'est-à-dire entre 0 et $+\infty$ pour les amplitudes et entre 0 et 2π pour les phases, et on omettra d'écrire ces bornes sur les intégrales pour rester concis. On pourra donc trouver les écritures compactes suivantes, pour f et g deux fonctions intégrables:

$$\begin{aligned}\int f(a)da &= \int \dots \int_{[0,+\infty)^N} f(a_1, \dots, a_N) da_1, \dots, da_N \\ \int g(\varphi)d\varphi &= \int \dots \int_{[0,2\pi)^N} g(\varphi_1, \dots, \varphi_N) d\varphi_1, \dots, d\varphi_N\end{aligned}$$

et le même principe d'écriture est appliqué lorsque l'on intègre sur a et φ simultanément.

Dans le **cas monocible**, cas particulier du cas multicible, une seule cible est présente, donc x , a et φ ne sont composés que d'un seul élément, on s'affranchit donc de l'indice j non nécessaire.

Définitions et expressions générales

Il est désormais envisageable de définir la vraisemblance. A partir du modèle d'observation, il est possible de déterminer la probabilité d'obtenir une mesure connaissant l'état de la ou des cibles, leurs amplitudes et leurs phases.

- C'est ce que l'on connaît, et est noté $p(z|x, a, \varphi)$ (développé au paragraphe 5.3.3)

Ici z représente l'image radar sur toutes les cellules. Les variables x , a et φ peuvent être multi-cibles. Une observation image z est composée d'observations z^c dans chaque cellule de résolution c . Souvent, les observations de chaque cellule, conditionnées par l'état des cibles, leurs amplitudes et leurs phases, sont considérées comme indépendantes (grâce à un échantillonnage judicieux du signal), ce qui permet d'écrire

$$p(z|x, a, \varphi) = \prod_c p(z^c|x, a, \varphi) \quad (5.21)$$

Il est nécessaire enfin de marginaliser la densité $p(z|x, a, \varphi)$ sur l'amplitude et la phase, afin de déterminer la vraisemblance d'une observation, conditionnellement au vecteur d'état seulement.

- C'est ce que l'on cherche, et est noté $p(z|x)$ (développé au paragraphe 5.3.3)

Cette marginalisation s'écrit

$$\begin{aligned} p(z|x) &= \int p(z, a, \varphi|x) da d\varphi \\ &= \int p(z|x, a, \varphi) p(a, \varphi|x) da d\varphi \end{aligned}$$

or a , φ et x sont indépendants, donc $p(a, \varphi|x) = p(a)p(\varphi)$. Dans le cas multi-cibles, les amplitudes et phases des cibles sont indépendantes, on a donc également $p(a) = \prod_{j=1}^N p(a_j)$ et $p(\varphi) = \prod_{j=1}^N p(\varphi_j)$ avec N le nombre de cibles, a_j et φ_j respectivement l'amplitude et la phase de la cible j . Donc

$$p(z|x) = \int p(z|x, a, \varphi) p(a) p(\varphi) da d\varphi \quad (5.22)$$

De nombreux travaux [BDV⁺03, DRC12, MB08b, KBB16] concernant la vraisemblance ont été réalisés, et largement étendus dans [LRL13, LRL16]. Ils regroupent des stratégies différentes.

Observation complexe Afin de ne pas perdre l'information apportée par la phase et au contraire l'exploiter, on peut considérer directement le signal complexe. Dans ce cas c'est

l'expression générale (5.22) qui est utilisée. Dans les travaux initiateurs [DRC12], seule la phase est complètement exploitée, suggérant ensuite seulement de réaliser l'intégration pour marginaliser sur l'amplitude. Les travaux [LRL16] s'attèlent donc à cette tâche, dans le cas monocible et le cas multicible afin de fournir des expressions explicites si possible, ou sinon des expressions alternatives, par exemple des expressions approchées.

Observation du module ou du module au carré

Etant donné un signal complexe z , on peut considérer le module $|z|$ de ce signal, ou le module $|z|^2$ mis au carré. Les raisonnements ci-après sont donnés pour $|z|^2$ mais sont également valables pour $|z|$, seules les expressions explicites diffèrent bien sûr. Exploiter le module permet donc de s'affranchir de la phase dans le cas monocible; le module ne dépend plus que de l'amplitude. Dans le cas multicible, cela permet de s'affranchir d'une seule phase, appelée phase de référence; le module dépend alors des amplitudes et des phases restantes, notées φ_- . Dans ce cas, sur le modèle de (5.22), la variable $Y = |z|^2$ a pour vraisemblance

$$p(y|x) = \int p(y|x, a, \varphi_-) p(a) p(\varphi_-) da d\varphi_- \quad \text{multicible} \quad (5.23)$$

$$p(y|x) = \int p(y|x, a) p(a) da \quad \text{monocible} \quad (5.24)$$

En fait l'expression multi-cible contient le cas monocible en définissant dans le cas monocible $\varphi_- = \emptyset$, c'est-à-dire l'ensemble des phases restantes est vide. En effet dans le cas monocible il n'y qu'une phase, celle de l'unique cible, et donc aucune autre phase restante. Cette remarque est valable pour les équations suivantes exprimées dans les deux cas, monocible et multicible.

Indépendance cellule à cellule de $z|x$ Sous l'hypothèse que les observations, conditionnées par l'état de la ou des cibles x , sont indépendantes de cellule en cellule, la loi jointe des observations, qui est la loi de l'observation image, peut s'écrire comme le produit des lois des observations de chacune des cellules [MB08b]. C'est-à-dire que la variable $Y = |z|^2$ a pour densité

$$p(y|x) = \prod_c p(y^c|x) \quad (5.25)$$

avec la variable $Y^c = |z^c|^2$ et z^c l'observation dans la cellule c , et

$$p(y^c|x) = \int p(y^c|x, a, \varphi_-) p(a) p(\varphi_-) da d\varphi_- \quad \text{multicible} \quad (5.26)$$

$$p(y^c|x) = \int p(y^c|x, a) p(a) da \quad \text{monocible} \quad (5.27)$$

Cette hypothèse diffère de l'hypothèse (5.21), car dans l'hypothèse (5.21) les observations sont bien considérées comme indépendante de cellule en cellule, mais conditionnées par l'état des cibles, leurs amplitudes et leurs phases. Dans la nouvelle hypothèse (5.25), les observations sont liées, puisque les amplitudes et les phases des cibles leur sont communes. C'est donc une hypothèse lourde de conséquences. L'intégrale (5.26) est calculée pour chaque cellule, sur laquelle on intègre sur l'amplitude, et les phases restantes, de façon indépendante. La phase de référence étant de fait écartée, on en perd toute l'information. Ainsi la cohérence spatiale de ces variables n'est pas exploitée dans cette stratégie.

Cohérence avec l'amplitude L'hypothèse (5.25) est écartée, ce sont donc les équations (5.23) et (5.24) qui sont directement appliquées.

Alternative à la marginalisation par intégration brute

La marginalisation sur les phases et les amplitudes, définie par l'expression (5.22), fait appel à la vraisemblance conditionnée par les phases et les amplitudes $p(z|x, a, \varphi)$. Cette vraisemblance conditionnée est dérivée à partir du modèle d'observation, grâce aux propriétés statistiques du bruit additif n . Cependant, le bruit n'est pas la seule variable aléatoire intervenant dans le modèle d'observation: les amplitudes et les phases ont également des propriétés statistiques. Dans certains cas précis il sera donc possible d'utiliser ces propriétés pour reconnaître des distributions associées et donc en déduire directement la distribution $p(z|x)$ sans avoir à résoudre l'intégrale multiple de marginalisation. Par exemple, si la variable X suit une loi de Rayleigh de paramètre σ et si la variable θ suit une loi uniforme sur $[0, 2\pi)$, alors la variable $Y = Xe^{i\theta}$ suit une loi complexe gaussienne de moyenne nulle et de variance $2\sigma^2$.

Vraisemblances conditionnées par x , a et φ

Déjà, il est possible de donner une forme explicite à ce que l'on connaît grâce à l'équation d'observation (5.15) et aux propriétés statistiques du bruit additif n .

Le bruit a pour moyenne 0 et pour matrice de covariance Σ_n , donc la variable d'observation z conditionnée par les états des cibles x , leurs amplitudes a et leurs phases φ , est également

une variable gaussienne complexe, de même covariance, mais de moyenne μ comme définie par (5.16), et donc on peut écrire

$$p(z|x, a, \varphi) = \frac{1}{\pi^{N_c} |\Sigma_n|} \exp(-(z - \mu)^H \Sigma_n^{-1} (z - \mu)) \quad (5.28)$$

où $|\cdot|$ dénote le déterminant d'une matrice et $(\cdot)^H$ le conjugué hermitien (c'est-à-dire le conjugué transposé). À noter cependant que si $|\cdot|$ est appliqué ailleurs dans le document à un nombre il s'agit du module de ce nombre, et si appliqué à un vecteur il s'agit du vecteur des modules des éléments.

Il est intéressant aussi de définir la densité de probabilité de l'observation dans le cas où aucune cible n'est présente; seul le bruit est observé et $z = n$. On appelle ce cas l'hypothèse nulle $\emptyset$.

$$p_{\emptyset}(z) = \frac{1}{\pi^{N_c} |\Sigma_n|} \exp(-z^H \Sigma_n^{-1} z) \quad (5.29)$$

Dans le cas où la covariance du bruit est diagonale, et s'écrit $\Sigma_n = \sigma_n^2 \times I_{N_c}$, l'hypothèse d'indépendance des observations entre chaque cellule, donnée par l'expression (5.21) est bien vérifiée, et le conditionnement est codé par l'expression de la moyenne μ en fonction de (x, a, φ) . On peut tout à fait réécrire alors les expressions ci-dessus pour chaque cellule:

$$(5.17) \Rightarrow z^c = \mu^c + n^c \quad (5.30)$$

$$(5.16) \Rightarrow \mu^c = \sum_{j=1}^N a_j e^{i\varphi_j} f^c(x_j) \quad (5.31)$$

$$(5.3.3) \Rightarrow p(z^c|x, a, \varphi) = \frac{1}{\pi \sigma_n^2} \exp\left(-\frac{1}{\sigma_n^2} |z^c - \mu^c|^2\right)$$

$$(5.29) \Rightarrow p_{\emptyset}(z^c) = \frac{1}{\pi \sigma_n^2} \exp\left(-\frac{1}{\sigma_n^2} |z^c|^2\right)$$

Lorsque l'on prend le module au carré de l'observation (complexe), l'équation d'observation (5.30) donne alors

$$\begin{aligned} |z^c|^2 &= \mathcal{Re}(z^c)^2 + \mathcal{Im}(z^c)^2 \\ &= \underbrace{(\mathcal{Re}(\mu^c) + \mathcal{Re}(n^c))^2}_{\mathcal{N}(\mathcal{Re}(\mu^c), \frac{\sigma_n^2}{2})} + \underbrace{(\mathcal{Im}(\mu^c) + \mathcal{Im}(n^c))^2}_{\mathcal{N}(\mathcal{Im}(\mu^c), \frac{\sigma_n^2}{2})} \end{aligned}$$

Or $\mathcal{Re}(n^c)$ et $\mathcal{Im}(n^c)$ sont toutes les deux des variables réelles suivant une loi gaussienne de moyenne nulle et de même variance $\sigma_n^2/2$. Donc chacun des termes (sans le carré) suit une loi

gaussienne de moyenne respectivement $\mathcal{Re}(\mu^c)$ et $\mathcal{Im}(\mu^c)$, et de même variance $\sigma_n^2/2$. Pour reconnaître une variable du khi-deux non centrée avec deux degrés de liberté (car somme de deux termes gaussiens), il faut réduire (diviser par l'écart type) les variables gaussiennes. Alors

$$\left(\frac{\mathcal{Re}(\mu^c) + \mathcal{Re}(n^c)}{\sqrt{\sigma_n^2/2}} \right)^2 + \left(\frac{\mathcal{Im}(\mu^c) + \mathcal{Im}(n^c)}{\sqrt{\sigma_n^2/2}} \right)^2 = \frac{2}{\sigma_n^2} |z^c|^2$$

suit une loi du khi-deux non centrée avec deux degrés de liberté, et de paramètre de décentralisation λ^c définit par

$$\begin{aligned} \lambda^c &= \left(\frac{\mathcal{Re}(\mu^c)}{\sqrt{\sigma_n^2/2}} \right)^2 + \left(\frac{\mathcal{Im}(\mu^c)}{\sqrt{\sigma_n^2/2}} \right)^2 \\ &= \frac{2}{\sigma_n^2} (\mathcal{Re}(\mu^c)^2 + \mathcal{Im}(\mu^c)^2) \\ &= \frac{2}{\sigma_n^2} |\mu^c|^2 \end{aligned} \tag{5.32}$$

On note alors

$$\frac{2}{\sigma_n^2} |z^c|^2 \sim \chi_2^2(\lambda^c)$$

Et donc, la variable $Y^c = |z^c|^2$ a pour densité

$$p(y^c|x, a, \varphi) = \frac{2}{\sigma_n^2} p_{\chi_2^2(\lambda^c)} \left(\frac{2}{\sigma_n^2} y^c \right) \tag{5.33}$$

avec $p_{\chi_2^2(\lambda)}(\cdot)$ la densité du khi-deux non centrée à deux degrés de liberté. Le conditionnement selon les états des cibles x , leurs amplitudes a et leurs phases φ est contenu dans l'expression de μ^c et donc de λ^c . En explicitant un peu plus le paramètre de décentralisation λ^c selon le cas monocible ou multicible on obtient:

- λ^c pour le cas monocible: il n'y a qu'une seule cible dont l'amplitude est a et la phase φ .
Donc à partir de (5.32)

$$\begin{aligned} \lambda_{\text{mono}}^c &= \frac{2}{\sigma_n^2} |\mu^c|^2 \\ &= \frac{2}{\sigma_n^2} |a e^{i\varphi} f^c(x)|^2 \\ &= \frac{2}{\sigma_n^2} a^2 f^c(x)^2 \\ &= \lambda^c(x, a) \end{aligned}$$

Donc le conditionnement par la phase φ disparaît puisqu'elle n'apparaît plus dans l'expression de la densité de probabilité.

- λ^c pour le cas multicible: il y a plusieurs cibles dont les amplitudes et les phases sont les composantes du vecteur a et du vecteur φ respectivement. Donc à partir de (5.32)

$$\begin{aligned}
 \lambda_{\text{multi}}^c &= \frac{2}{\sigma_n^2} |\mu^c|^2 \\
 &= \frac{2}{\sigma_n^2} \left| \sum_{j=1}^N a_j e^{i\varphi_j} f^c(x_j) \right|^2 \\
 &= \frac{2}{\sigma_n^2} \left| e^{-i\varphi_1} \left(a_1 f^c(x_1) + \sum_{j=2}^N a_j e^{i(\varphi_j - \varphi_1)} f^c(x_j) \right) \right|^2 \\
 &= \frac{2}{\sigma_n^2} \left| a_1 f^c(x_1) + \sum_{j=2}^N a_j e^{i\varphi'_j} f^c(x_j) \right|^2 \\
 &= \lambda^c(x, a, \varphi_-)
 \end{aligned}$$

avec $\varphi'_j = \varphi_j - \varphi_1$ pour $j = 2, \dots, N$, la phase relative, dont les propriétés statistiques, c'est-à-dire uniforme sur $[0, 2\pi)$, restent inchangées. Ainsi on s'affranchit de la dépendance d'une des phases, mais pas de celles des phases restantes notées φ_- . Le cas monocible est simplement un cas particulier du cas multicible dans lequel $\varphi_- = \emptyset$

Finalement, après ces considérations de conditionnement on peut donc écrire, en remplaçant $p_{\chi_2^2(\lambda^c)}(\cdot)$ par sa valeur dans (5.33), la variable $Y^c = |z^c|^2$ a pour densité

$$p(y^c|x, a) = \frac{1}{\sigma_n^2} \exp\left(-\left(\frac{y^c}{\sigma_n^2} + \frac{\lambda^c(x, a)}{2}\right)\right) I_0\left(\sqrt{\frac{2}{\sigma_n^2} y^c \lambda^c(x, a)}\right) \quad \text{monocible} \quad (5.34)$$

$$p(y^c|x, a, \varphi_-) = \frac{1}{\sigma_n^2} \exp\left(-\left(\frac{y^c}{\sigma_n^2} + \frac{\lambda^c(x, a, \varphi_-)}{2}\right)\right) I_0\left(\sqrt{\frac{2}{\sigma_n^2} y^c \lambda^c(x, a, \varphi_-)}\right) \quad \text{multicible} \quad (5.35)$$

avec I_0 la fonction de Bessel modifiée de première espèce.

De la même manière que pour le cas complexe avec l'équation (5.29), on peut définir la densité de probabilité de $Y^c = |z^c|^2$ dans le cas où aucune cible n'est présente (hypothèse nulle). Il s'agit d'une densité du khi-deux à deux degrés de liberté (mise à l'échelle), ou de

manière équivalente une distribution exponentielle de paramètre σ_n^2 , qui s'écrit alors

$$p_{\emptyset}(y^c) = \frac{1}{\sigma_n^2} \exp\left(-\frac{y^c}{\sigma_n^2}\right) \quad (5.36)$$

Remarque pour le cas Swerling 0 Dans le cas où les amplitudes des cibles suivent un modèle de fluctuation de Swerling 0, les amplitudes sont constantes et égales aux paramètres $\gamma_a = \{\gamma_{a_j}\}_{j=1}^N$, et chaque amplitude a une distribution définie par (5.18). La marginalisation selon les amplitudes consiste donc à remplacer les amplitudes dans l'expression que l'on intègre par leurs valeurs constantes. Cette remarque peut s'appliquer indifféremment avant ou après la marginalisation sur la ou les phases par exemple, et aussi bien sur l'observation complexe que sur l'observation du module au carré, sur l'image d'observation entière ou l'observation d'une cellule.

$$\begin{aligned} p(z|x, \varphi) &= p(z|x, a = \gamma_a, \varphi) \\ p(z|x) &= p(z|x, a = \gamma_a) \end{aligned}$$

Vraisemblances conditionnées par x , marginalisées selon a et φ

Désormais toutes les expressions qu'il est possible de déduire grâce aux propriétés statistiques du bruit sont déterminées. Les expressions non conditionnées par les paramètres inconnus que sont les amplitudes et les phases (ou phases restantes) sont donc à déterminer en marginalisant sur ces paramètres. Pour cela, on reprend les différentes stratégies énoncées aux paragraphes 5.3.3 et 5.3.3: observation du module au carré (indépendance cellule à cellule non conditionnée par a et φ ou cohérence avec l'amplitude) ou observation complexe. Ces stratégies sont combinées aux différents modèles de Swerling.

Le paramètre d'amplitude γ_a est à présent un paramètre à part entière du système. Il peut s'agir d'un scalaire dans le cas monocible, ou d'un vecteur dans le cas multicible. Dans le cas d'un modèle de Swerling 0 pour la fluctuation d'amplitude, γ_a donne la valeur constante de chacune des amplitudes, qui ne sont plus des variables aléatoires. Les valeurs des amplitudes peuvent donc directement être utilisées dans les équations définies ci-avant. Dans les autres cas des modèles de Swerling, γ_a donne le paramètre de la distribution de chacune des amplitudes, et intervient dans l'expression finale de la vraisemblance. Le paramètre peut être considéré comme connu. Sinon il peut être ajouté à l'état de chacune des cibles, c'est-à-dire être contenu dans x . Quand à la loi uniforme sur les phases, elle ne fait pas intervenir de paramètre supplémentaire.

Observation complexe

Swerling 0 Les cas monocible et multicible sont distingués, une expression explicite pouvant être dérivée pour le cas monocible, une approximation étant nécessaire pour le cas multicible. Pour rappel, avec le modèle de Swerling 0 les amplitudes des cibles sont constantes, égales aux paramètres γ_{a_j} . L'expression (5.16) devient alors $\mu = \sum_{j=1}^N \gamma_{a_j} e^{i\varphi_j} f(x_j)$. Seule reste la marginalisation selon les phases des cibles pour obtenir la vraisemblance d'intérêt. A partir de (5.3.3) on a

$$p(z|x, \varphi) = \frac{1}{\pi^{N_c} |\Sigma_n|} \exp(-(z - \mu)^H \Sigma_n^{-1} (z - \mu)) \quad (5.37)$$

Pour le **cas monocible**, il est nécessaire de marginaliser sur une seule phase φ . Il s'agit des résultats présentés dans l'article [DRC12]. Etant donné la présence d'une seule cible, l'expression (5.16) se réduit à $\mu = \gamma_a e^{i\varphi} f(x)$. Alors l'expression (5.37) une fois développée donne

$$p(z|x, \varphi) = p_{\emptyset}(z) \exp(2\operatorname{Re}(\mu^H \Sigma_n^{-1} z)) \exp(-\mu^H \Sigma_n^{-1} \mu) \quad (5.38)$$

Or

$$\begin{aligned} \mu^H \Sigma_n^{-1} z &= \gamma_a e^{-i\varphi} f(x)^T \Sigma_n^{-1} z \\ &= (\gamma_a f(x)^T \Sigma_n^{-1} z) e^{-i\varphi} \\ &= \gamma_a |f(x)^T \Sigma_n^{-1} z| e^{i(\theta - \varphi)} \end{aligned}$$

en définissant

$$\theta = \arg(\gamma_a f(x)^T \Sigma_n^{-1} z)$$

et $(\cdot)^T$ définit la transposée.

Donc finalement l'expression (5.39) devient

$$p(z|x, \varphi) = p_{\emptyset}(z) \exp(2\gamma_a |f(x)^T \Sigma_n^{-1} z| \cos(\theta - \varphi)) \exp(-\gamma_a^2 f(x)^T \Sigma_n^{-1} f(x))$$

Finalement en marginalisant sur la variable de la phase φ selon sa distribution uniforme on obtient [DRC12],

$$\begin{aligned} p(z|x) &= \int_0^{2\pi} \frac{1}{2\pi} p(z|x, \varphi) d\varphi \\ &= p_{\emptyset}(z) \exp(-\gamma_a^2 f(x)^T \Sigma_n^{-1} f(x)) I_0(2\gamma_a |f(x)^T \Sigma_n^{-1} z|) \end{aligned}$$

Pour le **cas multicible**, il est nécessaire de marginaliser sur les phases de toutes les cibles en présence, mais une expression explicite de cette intégrale n'est pas disponible. La solution retenue est d'utiliser l'approximation de Laplace de cette intégrale comme décrit dans [LRL16]. La marginalisation s'écrit alors à partir (5.37):

$$\begin{aligned} p(z|x) &= \int p(z|x, \varphi) \frac{1}{(2\pi)^N} d\varphi \\ &= \frac{1}{\pi^{N_c} |\Sigma_n|} \frac{1}{(2\pi)^N} \int \exp(-(z - \mu)^H \Sigma_n^{-1} (z - \mu)) d\varphi \end{aligned} \quad (5.39)$$

En notant $F = [\gamma_{a_1} f(x_1), \dots, \gamma_{a_N} f(x_N)]$ et $\Psi(\varphi) = [e^{i\varphi_1}, \dots, e^{i\varphi_N}]^T$, on peut alors écrire

$$\mu = F\Psi(\varphi)$$

et définir

$$\begin{aligned} g(\varphi) &= -(z - \mu)^H \Sigma_n^{-1} (z - \mu) \\ &= -(z - F\Psi(\varphi))^H \Sigma_n^{-1} (z - F\Psi(\varphi)) \end{aligned}$$

et ainsi réécrire (5.39)

La méthode de Laplace donne alors l'approximation de l'intégrale suivante

$$\int e^{g(\varphi)} d\varphi \approx (2\pi)^{N/2} | -H(g)(\hat{\varphi}) |^{-1/2} \exp(g(\hat{\varphi})) \quad (5.40)$$

avec $\hat{\varphi} = \operatorname{argmax}_{\varphi} g(\varphi)$ et $H(g)(\hat{\varphi})$ la matrice hessienne de g évaluée en $\hat{\varphi}$. Afin de déterminer $\hat{\varphi}$, deux méthodes sont proposées. La première est la méthode de descente de gradient. La seconde repose sur le fait de trouver $\Psi(\varphi)$ qui maximise $g(\varphi)$, en remarquant que $g(\varphi)$ est une forme quadratique en $\Psi(\varphi)$, et ainsi pour maximiser $g(\varphi)$ (et donc minimiser $-g(\varphi)$) il est possible d'utiliser la méthode des moindres carrés généralisées, dont l'estimateur possède alors la forme explicite suivante

$$\hat{\Psi}(\hat{\varphi}) = (F^T \Sigma_n^{-1} F)^{-1} F^T \Sigma_n^{-1} z \quad (5.41)$$

et ensuite on définit $\hat{\varphi} = \arg(\hat{\Psi}(\hat{\varphi}))$. Dans l'article [LRL16], les auteurs mettent en garde sur le fait que le vecteur Ψ calculé par l'expression (5.41) n'est pas un vecteur de phase dont les modules des éléments sont égaux à un, ce qui peut générer des erreurs.

Finalement, la vraisemblance (5.39) peut être approchée en injectant l'approximation de

Laplace (5.40) dans l'expression de la vraisemblance (5.39).

Swerling 1 et 2 Ici les cas monocible et multicible sont traités de la même manière, le cas monocible étant un cas particulier du cas multicible.

Chaque amplitude suit une loi de Rayleigh, et chaque phase une loi uniforme sur $[0, 2\pi)$, donc chaque variable $a_j e^{i\varphi_j}$ suit une loi gaussienne complexe circulaire symétrique de moyenne nulle et de variance γ_{a_j} . Par transformation linéaire, chaque variable $a_j e^{i\varphi_j} f(x_j)$ suit également une loi gaussienne complexe circulaire symétrique de moyenne nulle et de variance $\gamma_{a_j} f(x_j) f(x_j)^T$. Ainsi, l'observation définie par l'équation (5.15) est également une variable gaussienne complexe circulaire symétrique de moyenne nulle et de matrice de covariance

$$Q = \sum_{j=1}^N \gamma_{a_j} f(x_j) f(x_j)^T + \Sigma_n$$

La distribution de l'observation complexe s'écrit alors

$$p(z|x) = \frac{1}{\pi^{N_c} |Q|} \exp(-z^H Q^{-1} z) \quad (5.42)$$

Calculer l'inverse et le déterminant de Q peut demander une grande puissance de calcul. Il est cependant possible d'utiliser les lemmes d'algèbre linéaire concernant l'inversion de matrice (identité de Woodbury) et le déterminant de matrice, en écrivant Q sous la forme suivante

$$Q = F D F^T + \Sigma_n$$

avec $F = [f(x_1), \dots, f(x_n)]$ et $D = \text{Diag}(\gamma_{a_1}, \dots, \gamma_{a_n})$. On a alors

$$Q^{-1} = \Sigma_n^{-1} - \underbrace{\Sigma_n^{-1} F (D^{-1} + F^T \Sigma_n^{-1} F)^{-1} F^T \Sigma_n^{-1}}_M$$

$$|Q| = \underbrace{|D^{-1} + F^T \Sigma_n^{-1} F|}_d \times |D| \times |\Sigma_n|$$

Et on peut finalement réécrire (5.42)

$$p(z|x) = p_\varnothing(z) \frac{1}{d} \exp(z^H M z)$$

Swerling 3 et 4 Pour les modèles de Swerling 3 et 4, les cas monocible et multicible sont distingués et traités séparément. En effet dans le cas monocible une expression explicite peut être dérivée, ce qui n'est pas le cas du multicible pour lequel une alternative sera envisagée.

Dans le **cas monocible**, le développement de l'expression avec $\mu = ae^{i\varphi}f(x)$ donne

$$p(z|x, a, \varphi) = p_{\emptyset}(z) \exp(2a|f(x)^T \Sigma_n^{-1} z| \cos(\theta - \varphi)) \exp(-a^2 f(x)^T \Sigma_n^{-1} f(x))$$

qui une fois marginalisée sur la seule phase donne

$$p(z|x, a) = p_{\emptyset}(z) \exp(-a^2 f(x)^T \Sigma_n^{-1} f(x)) I_0(2a|f(x)^T \Sigma_n^{-1} z|) \quad (5.43)$$

Il reste donc la marginalisation sur la seule amplitude selon le modèle de Swerling 3 et 4, dont l'intégrale correspondante s'écrit alors

$$\begin{aligned} p(z|x) &= \int \underbrace{p(z|x, a)}_{(5.43)} \underbrace{p(a)}_{(5.20)} da \\ &= p_{\emptyset}(z) \frac{8}{\gamma_a^2} \int a^3 \exp\left(-\left(\frac{2}{\gamma_a} + f(x)^T \Sigma_n^{-1} f(x)\right) a^2\right) I_0(2|f(x)^T \Sigma_n^{-1} z|a) da \end{aligned}$$

et possède une solution explicite déduite des tables d'intégrales [GR07] (p. 709, section 6.643, équation 2.). On obtient finalement

$$p(z|x) = p_{\emptyset}(z) \frac{4}{(2 + \gamma_a f(x)^T \Sigma_n^{-1} f(x))^2} \left(\frac{\gamma_a |f(x)^T \Sigma_n^{-1} z|^2}{2 + \gamma_a f(x)^T \Sigma_n^{-1} f(x)} + 1 \right) \exp\left(\frac{\gamma_a |f(x)^T \Sigma_n^{-1} z|^2}{2 + \gamma_a f(x)^T \Sigma_n^{-1} f(x)} \right)$$

Pour le **cas multicible**, afin de déterminer la vraisemblance, la distribution du khi à quatre de degrés de liberté (modèle de Swerling 3 et 4 défini au paragraphe 5.3.2) va être approximée par une distribution de Rice, pour le modèle de fluctuation de l'amplitude. Ainsi la densité de probabilité de l'amplitude selon la distribution de Rice $\left(\sigma = \sqrt{\frac{\gamma_a}{2(1+\alpha^2)}}, \nu = \alpha \sqrt{\frac{\gamma_a}{(1+\alpha^2)}}\right)$ est définie par l'expression suivante:

$$p(a) = \frac{2(1 + \alpha^2)}{\gamma_a} a \exp\left(-\alpha^2 - \frac{1 + \alpha^2}{\gamma_a} a^2\right) I_0\left(2\alpha \sqrt{\frac{1 + \alpha^2}{\gamma_a}} a\right)$$

En choisissant $\alpha = \sqrt{1 + \sqrt{2}}$, les densités de l'amplitude au carré selon le modèle de Rice défini ci-dessus ou le modèle de Swerling 3 et 4, ont même moyenne et même espérance. Chaque variable $a_j e^{i\varphi_j}$ suit alors une distribution gaussienne complexe $\mathcal{CN}(\nu_j e^{i\theta_j}, 2\sigma_j^2)$, conditionnée par θ_j variable uniforme sur $[0, 2\pi)$. L'observation est alors la somme de variables gaussi-

ennes complexes, conditionnées par les variables⁴ θ , et est donc une gaussienne complexe de moyenne μ et matrice de covariance Q ,

$$\mu = \sum_{j=1}^N \nu_j e^{i\theta_j} f(x_j)$$

$$Q = \Sigma_n + \sum_{j=1}^N 2\sigma_j^2 f(x_j) f^T(x_j)$$

avec

$$\nu_j = \alpha \sqrt{\frac{\gamma_{a_j}}{(1 + \alpha^2)}}$$

$$\sigma_j = \sqrt{\frac{\gamma_{a_j}}{2(1 + \alpha^2)}}$$

Finalement, on obtient l'expression de la vraisemblance, conditionnée par les variables θ , suivante

$$p(z|x, \theta) = \frac{1}{\pi^{N_c} |Q|} \exp(-(z - \mu)^H Q^{-1} (z - \mu))$$

Il reste ensuite à intégrer numériquement sur les variables θ uniformes sur $[0, 2\pi)$. Pour ce faire, à la place d'une intégration de Monte Carlo, une stratégie déterministe d'intégration sur grille fixe uniforme est adoptée, couvrant de façon plus systématique le domaine joint $[0, 2\pi)^N$ des variables θ . Cette stratégie permet d'utiliser moins de particules et ainsi limiter le coût de calcul.

Module au carré: Indépendance cellule à cellule de $z|x$ (non cohérent)

Dans ce cas les étapes sont les suivantes:

1. Calculer $p(|z^c|^2|x)$ avec (5.26) ou (5.27)
2. Réaliser le produit sur les cellules (5.25)

L'étape 1 doit être menée selon le modèle de Swerling choisi. Il s'agit donc de réaliser l'intégration sur chacune des cellules. Ainsi on intègre les amplitudes et les phases indépendamment sur chaque cellule alors qu'il s'agit normalement d'une valeur partagée. C'est donc ici une hypothèse forte.

⁴on notera les variables $\{\theta_j\}_{j=1}^N = \theta$

Swerling 0 Dans le **cas monocible**, il s'agit donc d'intégrer l'équation (5.34) sur l'amplitude dont la densité est (5.18).

$$p(|z^c|^2|x) = \int p(|z^c|^2|x, a) \delta(a - \gamma_a) da = p(|z^c|^2|x, \gamma_a)$$

il suffit donc d'insérer dans l'expression (5.34) la valeur de l'amplitude.

Pour le **cas multicible**, le raisonnement pour l'amplitude est similaire, il reste cependant à procéder à l'intégration sur les phases restantes φ_- . Cependant, il est plus judicieux d'utiliser le cas avec la cohérence d'amplitude comme il sera développé ci-après.

Swerling 1 et 2 Ici le **cas monocible** se déduit directement du **cas multicible**. On pourrait appliquer l'intégration de (5.35) sur l'amplitude dont la densité est (5.19) et sur les phases restantes. Cependant, on peut reprendre le raisonnement à partir des équations (5.30) et (5.31). En effet l'amplitude de chaque cible j suit une distribution de Rayleigh de paramètre $\sqrt{\gamma_a/2}$, et la phase de chaque cible j suit une distribution uniforme sur $[0, 2\pi)$, donc chaque variable $a_j e^{i\varphi_j}$ est une variable gaussienne complexe circulaire symétrique de moyenne nulle et de variance γ_{a_j} . Par transformation linéaire, chaque variable $a_j e^{i\varphi_j} f^c(x_j)$ est une variable gaussienne complexe circulaire symétrique de moyenne nulle et de variance $\gamma_{a_j} f^c(x_j)^2$. L'observation complexe z^c sur la cellule c est donc, d'après (5.30) et (5.31), la somme de variables gaussiennes complexes circulaires symétriques et du bruit lui-même gaussien complexe circulaire symétrique. L'observation complexe z^c est donc une variable gaussienne complexe circulaire symétrique, de moyenne nulle et de variance

$$\sigma_{z^c}^2 = \sigma_n^2 + \sum_{j=1}^N \gamma_{a_j} f^c(x_j)^2 \quad (5.44)$$

Cette distribution se note:

$$z^c \sim \mathcal{CN}(0, \sigma_{z^c}^2)$$

Finalement, la variable $\frac{2}{\sigma_{z^c}^2} |z^c|^2$ suit une loi du khi-deux, donc $|z^c|^2$ une loi du khi-deux (mise à l'échelle) ou de façon équivalente une loi exponentielle de paramètre d'espérance $\sigma_{z^c}^2$ défini par (5.44). Ainsi, en notant $Y^c = |z^c|^2$, on écrit

$$p(y^c|x) = \frac{1}{\sigma_{z^c}^2} e^{-\frac{y^c}{\sigma_{z^c}^2}}$$

Et finalement la vraisemblance (non cohérente spatialement) jointe sur l'image observation

est donné par le produit de l'expression (5.25).

Swerling 3 et 4 Pour les modèles de Swerling 3 et 4, les cas monocible et multicible sont distingués et traités séparément. En effet dans le cas monocible une expression explicite peut être dérivée, ce qui n'est pas le cas du multicible pour lequel une alternative sera envisagée.

Dans le **cas monocible**, afin de définir la vraisemblance du module au carré de l'observation sur une cellule $|z^c|^2$, une expression explicite va pouvoir être déduite de la marginalisation donnée par l'expression (5.27) avec la vraisemblance conditionnée par l'amplitude donnée par l'expression (5.34) et la distribution de l'amplitude définie par (5.20). Donc on cherche à déterminer, en notant la variable $Y^c = |z^c|^2$,

$$\begin{aligned} p(y^c|x) &= \int \frac{1}{\sigma_n^2} \exp\left(-\left(\frac{y^c}{\sigma_n^2} + \frac{\lambda^c(x,a)}{2}\right)\right) I_0\left(\sqrt{\frac{2}{\sigma_n^2}} y^c \lambda^c(x,a)\right) \frac{8a^3}{\gamma_a^2} \exp\left(-\frac{2a^2}{\gamma_a}\right) da \\ &= \underbrace{\frac{1}{\sigma_n^2} \exp\left(-\frac{y^c}{\sigma_n^2}\right)}_{p_\emptyset(y^c) \text{ (5.36)}} \frac{8}{\gamma_a^2} \int a^3 \exp\left(-\left(\frac{f^c(x)^2}{\sigma_n^2} + \frac{2}{\gamma_a}\right) a^2\right) I_0\left(\sqrt{\frac{2}{\sigma_n^2}} y^c \lambda^c(x,a)\right) da \end{aligned}$$

En utilisant le résultat suivant :

$$\int_0^{+\infty} x^3 e^{-\alpha x^2} I_0(\beta x) dx = \frac{1}{2\alpha^2} \left(\frac{\beta^2}{2\alpha} + 1\right) \exp\left(\frac{\beta^2}{4\alpha}\right)$$

on obtient finalement,

$$p(y^c|x) = p_\emptyset(y^c) \frac{(2\sigma_n^2)^2}{(2\sigma_n^2 + \gamma_a |f^c(x)|^2)^2} \left(1 + \frac{\gamma_a |f^c(x)|^2 y^c}{\sigma_n^2 (2\sigma_n^2 + \gamma_a |f^c(x)|^2)}\right) \exp\left(\frac{\gamma_a |f^c(x)|^2 y^c}{\sigma_n^2 (2\sigma_n^2 + \gamma_a |f^c(x)|^2)}\right)$$

Pour le **cas multicible**, afin de déterminer la vraisemblance, la même approximation pour la fluctuation de l'amplitude qu'à la p.132 est appliquée. C'est-à-dire que la distribution initiale de Swerling 3 et 4 est approximée par une distribution de Rice. Dans cette section, les cellules sont considérées comme indépendantes les unes des autres. Ce sont donc les expressions (5.30) et (5.31) qui sont utilisées comme point de départ. L'observation complexe de chaque cellule est donc une gaussienne complexe, conditionnées par les variables θ , de moyenne μ^c et variance q^c .

$$\mu^c = \sum_{j=1}^N \nu_j e^{i\theta_j} f^c(x_j)$$

$$q^c = \sigma_n^2 + \sum_{j=1}^N 2\sigma_j^2 (f^c(x_j))^2$$

avec chaque variable θ_j uniforme sur $[0, 2\pi)$. Le module au carré de l'observation complexe de chaque cellule aura donc une distribution du khi-deux non centrée à deux degrés de liberté (mise à l'échelle). En effet, la variable $\frac{2}{q^c} |z^c|^2$ suit une distribution du khi-deux décentrée à deux degrés de liberté dont le paramètre de décentralisation est

$$\begin{aligned} \lambda^c &= \frac{2}{q^c} |\mu^c|^2 \\ &= \frac{2}{q^c} \left| \nu_1 f^c(x_1) + \sum_{j=2}^N \nu_j e^{i\theta'_j} f^c(x_j) \right|^2 \end{aligned}$$

avec $\theta'_j = \theta_j - \theta_1$. Les variables θ restantes sont notées $\theta_- = \{\theta_j\}_{j=2}^N$. Finalement, l'expression de la vraisemblance pour la variable $Y^c = |z^c|^2$, conditionnée par θ_- s'écrit alors

$$p(y^c | x, \theta_-) = \frac{1}{q^c} \exp \left(-\frac{x}{q^c} - \frac{\lambda^c}{2} \right) I_0 \left(\sqrt{\frac{2}{q^c} y^c \lambda^c} \right)$$

Il reste à intégrer numériquement sur les variables θ_- afin d'obtenir la vraisemblance d'intérêt $p(y^c | x)$.

Module au carré: Cohérence avec l'amplitude Dans ce cas les étapes sont les suivantes:

1. Réaliser le produit sur les cellules (5.21)
2. Calculer $p(|z|^2 | x)$ avec (5.23) ou (5.24)

Le produit sur les cellules de l'expression (5.35) (ou (5.34) dans le cas monocible) donne, en notant les variables $Y = |z|^2$ et $Y^c = |z^c|^2$

$$\begin{aligned} p(y | x, a, \varphi_-) &= \prod_{c=1}^{N_c} p(y^c | x, a, \varphi_-) \\ &= \prod_{c=1}^{N_c} p_{\varnothing}(y^c) \exp \left(-\frac{\lambda^c(x, a, \varphi_-)}{2} \right) I_0 \left(\sqrt{\frac{2}{\sigma_n^2} y^c \lambda^c(x, a, \varphi_-)} \right) \end{aligned} \quad (5.45)$$

le cas monocible étant un cas particulier du cas multicible où $\lambda^c(x, a, \varphi_-)$ est remplacé par $\lambda^c(x, a)$. L'étape 2 est ensuite à réaliser selon le modèle de Swerling choisi, c'est-à-dire la

marginalisation sur les variables a et φ_- . Dans le cas monocible, il n'y a aucune phase restante sur laquelle marginaliser, et il y a une seule amplitude à marginaliser; pour cela, une intégration numérique (de Monte Carlo) est proposée. Pour le cas multicible avec plus d'une cible, il est nécessaire de marginaliser sur plusieurs variables: les phases restantes (au moins une) et les amplitudes (au moins deux); l'intégration numérique n'est pas retenue car le coût de calcul peut s'avérer élevé. La solution proposée par [LRL16] consiste à remplacer le paramètre de décentralisation $\lambda^c(x, a, \varphi_-)$ par son espérance notée E_{λ^c}

$$E_{\lambda^c} = \frac{2}{\sigma_n^2} \sum_{j=1}^N \mathbb{E} [a_j^2] f^c(x_j)^2$$

donnant alors l'expression

$$p(y|x) = \prod_{c=1}^{N_c} p_{\emptyset}(y^c) \exp\left(-\frac{E_{\lambda^c}}{2}\right) I_0\left(\sqrt{\frac{2}{\sigma_n^2} y^c E_{\lambda^c}}\right) \quad (5.46)$$

Swerling 0 L'amplitude de chacune des cibles étant constante, aucune marginalisation n'est nécessaire sur les variables amplitudes. On remplace donc les amplitudes par leurs valeurs dans l'expression donnée ci-dessus (5.45). Dans le cas monocible on obtient donc directement l'expression suivante

$$p(y|x) = \prod_{c=1}^{N_c} p_{\emptyset}(y^c) \exp\left(-\frac{\lambda^c(x, \gamma_a)}{2}\right) I_0\left(\sqrt{\frac{2}{\sigma_n^2} y^c \lambda^c(x, \gamma_a)}\right)$$

Concernant le cas multicible, c'est l'expression (5.46) qui est utilisée, avec ici $\mathbb{E} [a_j^2] = \gamma_{a_j}^2$ dans l'expression de E_{λ^c} , selon la définition du modèle de Swerling 0.

Swerling 1 et 2 / Swerling 3 et 4 Dans le cas monocible, il est nécessaire de marginaliser sur une seule amplitude. Il est donc intéressant de réaliser une intégration de Monte Carlo, selon le modèle de Swerling correspondant.

$$\left\{a_j^{(i)}\right\}_{i=1}^{N_{MC}} \text{ avec } a_j \sim \text{Swerling}$$

$$p(y|x) \approx \frac{1}{N_{MC}} \sum_{i=1}^{N_{MC}} \prod_{c=1}^{N_c} p_{\emptyset}(y^c) \exp\left(-\frac{\lambda^c(x, a_j^{(i)})}{2}\right) I_0\left(\sqrt{\frac{2}{\sigma_n^2} y^c \lambda^c(x, a_j^{(i)})}\right)$$

Concernant le cas multicible, c'est l'expression (5.46) qui est utilisée, avec ici $\mathbb{E} [a_j^2] = \gamma_{a_j}^2$ dans l'expression de E_{λ^c} , selon la définition des modèles de Swerling correspondants.

			Swerling 0	Swerling 1	Swerling 3
Complexe	pleinement cohérent	mono	Marginalisation de la phase	Forme gaussienne	Marginalisation de la phase et de l'amplitude Distribution de Rice
		multi	Approximation de Laplace	Forme gaussienne	approximant Swerling 3 Intégration de MC sur les phases
Module au carré (non cohérent avec une phase)	non cohérent (avec l'amplitude)	mono	Marginalisation de la phase *	Forme exponentielle	Marginalisation de la phase et de l'amplitude Distribution de Rice
		multi	*	Forme exponentielle	approximant Swerling 3 Intégration de MC sur les phases
	cohérent (avec l'amplitude)	mono	Marginalisation de la phase	Marginalisation de la phase Intégration de MC sur l'amplitude	Marginalisation de la phase Intégration de MC sur l'amplitude
		multi	Loi du χ^2_2 non centrée	Loi du χ^2_2 non centrée Espérance du paramètre de décentralisation	Loi du χ^2_2 non centrée Espérance du paramètre de décentralisation

Table 5.2: Calculs des vraisemblances. ■ Forme explicite ■ Approximation numérique ■ Approximations probabiliste et numérique ■ Approximation heuristique ■ Approximation de Laplace. Les expressions avec * ne sont pas utilisées en pratique car leur équivalent cohérent peut être aussi facilement calculé. Le terme marginalisation signifie que l'intégration directe a une expression.

5.4 Gating (fenêtrage)

Introduction

Le *measurement gating*, ou simplement *gating*, masquage ou fenêtrage, est une stratégie mise en place afin de réduire la dimension de l'observation et ainsi limiter le coût de calcul, en écartant les observations qui ne semblent pas contenir d'information sur les cibles pour le problème de pistage. En effet, en TBD, l'image radar est constituée de très nombreuses cellules et le traitement d'une telle image peut s'avérer lourd, notamment dans les calculs de vraisemblances dans lesquels l'observation intervient. Cette stratégie apporte également une meilleure stabilité numérique des calculs, la grande dimension de l'espace d'observation pouvant donner de très petites valeurs pour le calcul de vraisemblance.

La fenêtre de validation, *gate* ou masque, est donc une sélection des observations. Ici il s'agit d'une zone définie à l'intérieur de laquelle les observations sont conservées, celles en dehors étant écartées pour le traitement du pistage. La fenêtre se définit à partir de l'estimation de l'état du système (que sont la ou les cibles), à partir de laquelle est déterminée une observation prédite, ou bien même une estimation de l'observation *a priori* si possible.

Le *gating* est à effectuer à chaque scan et pour chaque cible (ou piste) puisque que la fenêtre de validation peut dépendre de la précision avec laquelle l'état de la piste est connu [MM04].

Gating individuel et *gating* global

Lorsqu'une seule cible peut possiblement exister, et que le filtre utilisé n'a qu'une seule représentation (ou hypothèse) pour cette cible (par exemple une seule gaussienne résumant toute la connaissance statistique à propos de cette cible), alors le *gating* n'est établi qu'à partir de cette hypothèse. Lorsque plusieurs cibles sont susceptibles de coexister, ou bien qu'une cible a plusieurs représentations (mélange de gaussiennes, filtre particulière), ou les deux cas réunis, les fenêtres peuvent être définies individuellement sur chaque représentation, et la fenêtre globale est alors l'union des fenêtres individuelles.

Plutôt que de définir la fenêtre sur chacune des représentations disponibles pour une cible, une unique représentation de la cible peut être définie (une approximation gaussienne par exemple), sur laquelle est défini une unique fenêtre. Si plusieurs cibles coexistent, à nouveau la fenêtre globale est l'union des fenêtres de toutes les cibles.

Le *gating* dans le pistage sur données seuillées

Le *gating* est également appliqué pour du pistage de cibles sur des données ponctuelles (aussi appelées données seuillées, points de détection, plots, ...). Dans ce cadre, les observations peuvent provenir des cibles, mais également du bruit de l'équipement, du reflet de la mer, ou d'autres objets. Ces observations non souhaitables forment le *clutter*. Le *gating* consiste à ne

garder que les observations considérées comme proches, selon un critère défini, des observations prédites pour les hypothèses des cibles. Les observations ainsi conservées servent au traitement du pistage des cibles, les autres observations sont écartées. Le nombre d'observations étant ainsi réduit, les affectations entre les observations et les pistes sont réduites et le coût de calcul également. Le *gating* permet donc d'éliminer une partie des observations de *clutter* et ainsi d'en limiter les effets.

Typiquement la fenêtre va être définie comme une ellipsoïde autour de l'observation prédite de l'hypothèse de la cible. L'ellipsoïde est réduit à un intervalle en une dimension ou une ellipse en deux dimensions. L'ellipsoïde suppose une forme gaussienne de la distribution de l'observation autour de l'observation prédite de l'hypothèse de la cible. La taille de l'ellipsoïde correspond à une distance maximale (ou seuil) choisie à laquelle les observations doivent être de l'observation prédite (moyenne de la distribution gaussienne) pour être conservées. La distance d'un point à la distribution gaussienne est la distance de Mahalanobis. Si la distance de Mahalanobis d'un point est inférieure au seuil choisi, le point est à l'intérieur de l'ellipsoïde et appartient à la fenêtre. Inversement, si cette distance est supérieure au seuil, le point est en dehors de l'ellipsoïde, et le point n'appartient pas à la fenêtre. Le seuil, donnant la taille de l'ellipsoïde, dépend de la probabilité p_G que l'observation issue de la cible appartienne à l'ellipsoïde (et donc à la fenêtre). Par construction, la distance de Mahalanobis suit une loi du khi à deux degrés de liberté, permettant à partir de la probabilité p_G désirée de trouver le seuil définissant l'ellipsoïde de confiance.

Pour déterminer la fenêtre, la densité de probabilité de l'observation peut être exprimée comme suit [MM04]:

$$p(z_t|z_{0:t-1}) = \int p(z_t|x_t)p(x_t|z_{0:t-1})dx_t \quad (5.47)$$

Dans le cas d'un filtre particulaire, un ensemble de particules pondérées $\{w_{t-1}^i, x_{t-1}^i\}_{i=1}^{N_p}$ est disponible au temps précédent $t - 1$. A partir de ces particules, il est souhaitable d'obtenir rapidement une idée des nouvelles observations afin de déterminer le *gating*. Un choix rapide peut être de tirer des nouvelles particules selon la loi dynamique des cibles, c'est-à-dire $x_t^i \sim p(x_t|x_{t-1}^i)$, et alors l'expression (5.47) devient

$$p(z_t|z_{0:t-1}) \approx \sum_{i=1}^{N_p} w_{t-1}^i p(z_t|x_t^i) \quad (5.48)$$

Si précisément le modèle d'observation est de la forme

$$z_t = h(x_t) + e_t$$

avec $h(\cdot)$ la fonction d'observation et $e_t \sim \mathcal{N}(0, R)$, alors la vraisemblance $p(z_t|x_t)$ est gaussienne ($\mathcal{N}(z_t; h(x_t), R)$), et l'expression (5.48) est un mélange de gaussiennes. Une fenêtre peut alors être définie selon l'ellipsoïde précédemment introduite pour chaque élément i du mélange. La fenêtre globale sera alors l'union de chaque fenêtre i . Une alternative proposée dans [MM04] consiste à prendre une approximation gaussienne de (5.48), et de n'avoir à déterminer qu'une fenêtre globale directement.

Le *gating* dans le pistage TBD

Réduire la dimension de l'observation en TBD, appliqué au radar, revient à ne garder que les cellules informatives de l'observation, c'est-à-dire les cellules impactées par la présence d'une ou plusieurs cibles. L'impact des cibles sur l'image radar dépend de l'état des cibles et du modèle d'observation, et notamment de la fonction équipement du radar.

La stratégie de *gating* nécessite de définir

- une statistique sur l'état des cibles,
- une représentation des cibles dans l'espace d'observation,
- un critère ou une forme pour le masque afin de sélectionner les cellules de l'observation impactées par les cibles.

Stratégies existantes de *gating* en TBD avec filtre de type particulaire

Les stratégies existantes de *gating* vues en TBD consistent

- soit à sélectionner un nombre défini et constant de cellules autour de l'état estimé des cibles présentes [Lep16],
- soit à utiliser un critère de distance de Mahalanobis entre le centre des cellules et le nuage de particules représentant l'état des cibles présentes [Boc13].

La première solution ne prend pas en compte la variance de l'approximation particulaire. La deuxième solution revient à faire une approximation gaussienne du nuage de particules dans laquelle les cibles sont indépendantes (pas de corrélation entre les cibles). La distance de Mahalanobis permet d'évaluer la distance entre un point (ici le centre d'une cellule de l'observation) et une distribution dont on a établi la moyenne et la matrice de covariance (issues de l'approximation gaussienne du nuage de particules). Si la distance est en dessous d'un certain seuil, la cellule est sélectionnée pour le *gating*.

Problème des stratégies existantes

Les deux stratégies présentées ci-avant considèrent cependant que la présence d'une cible n'impacte que les cellules à proximité de la cible. Cela est vrai dans la mesure où le capteur radar et le traitement de son signal ont été établis afin d'obtenir une réponse maximum pour l'état vrai de la cible. Cependant ces traitements forment autour de la cible un motif étendu, représenté par la fonction d'équipement du capteur. Notamment la formations de lobes secondaires pour le traitement angulaire peut être prise en considération. Les cellules contenant les lobes secondaires sont alors une source d'information dans une stratégie TBD, alors qu'ils sont source de *clutter* et donc de fausses alarmes dans les stratégies par détection et données ponctuelles. De plus, si le nuage de particules est moins étendu que le motif du radar, alors l'observation risque d'être très réduite et l'information qu'elle contient avec.

Stratégie originale de *gating*: prise en compte de l'étalement de l'observation

L'idée développée ici s'appuie en partie sur la deuxième solution présentée ci-dessus, en proposant une approximation gaussienne pour chacune des cibles, afin de prendre en compte la variance de l'estimation. Plutôt que de ne garder que les cellules dont le centre est à une distance de Mahalanobis inférieure à un certain seuil formant une ellipsoïde, une sélection différente des cellules est proposée afin de prendre en compte l'impact des cibles sur l'image d'observation.

Application à l'image radar 2D

Le signal radar donnant une image en distance et en angle est étudié précédemment [réf de la section].

En **angle**, un lobe principal et des lobes secondaires représentent l'étalement de la cible. Le lobe principal est d'intensité plus importante que les lobes secondaires, cependant ces derniers sont bien présents, et ce sur tout l'intervalle angulaire considéré. De plus selon la fréquence angulaire à laquelle les observations sont prises, plus ou moins de cellules peuvent être dans le lobe principal: l'étalement angulaire de la cible peut être de plusieurs cellules. Le choix ici est de garder tous les angles, et donc de ne pas réaliser de *gating* angulaire.

En **distance**, l'étalement de la cible est défini par le filtre adapté. Celui-ci donne une réponse nulle sur certaines zones en distance. Il est donc tout à fait envisageable d'évincer les cellules qui sont dans cette zone et ne donnent aucune information sur la cible. Selon la forme du filtre adapté et la fréquence d'échantillonnage choisies, l'étalement en distance est souvent assez restreint.

Un nuage de particules multicibles pondérées $\{w_{t-1}^i, X_{t-1}^i\}_{i=1}^{N_p}$ au temps précédent $t-1$ est à disposition, lorsqu'une nouvelle observation z_t est disponible à l'instant t . Ici, comme seule la position des cibles intervient dans l'observation, c'est sur une prédiction de la position des

cibles que le *gating* va pouvoir être déterminé. La position est notée $\mathbf{p} = [x, y]$, et la position de la cible j est alors indicée $\mathbf{p}_j$. La recherche des cellules à conserver est faite pour chacune des cibles, qui sont considérées indépendantes pour ce traitement, et sont indépendantes dynamiquement. Une approximation gaussienne du nuage de particules prédit est alors envisagée. L'objectif est ici d'obtenir rapidement un aperçu de l'état prédit des cibles et de leur observation, afin d'obtenir la fenêtre de validation. Une fois obtenue, la fenêtre permettra de mettre en œuvre des techniques plus sophistiquées pour l'échantillonnage des particules. Un choix simple et rapide pour mener le *gating* peut donc être de définir l'approximation gaussienne en position de la cible j au pas de temps t par:

- l'estimation de la moyenne pondérée

$$\bar{\mathbf{p}}_{t|t-1,j} = \sum_{i=1}^{N_p} w_{t-1}^i \hat{\mathbf{p}}_{t|t-1,j}^i$$

avec $\hat{\mathbf{p}}_{t|t-1,j}^i$ la partie position de l'état prédit $\hat{x}_{t|t-1,j}^i$ de la cible j et de la particule i , pouvant par exemple être défini par

$$\hat{x}_{t|t-1,j}^i = \mathbb{E} [x_{t,j} | x_{t-1,j}^i] \quad (5.49)$$

- l'estimation non biaisée de la matrice de covariance pondérée,

$$\Sigma_{t,j} = \frac{1}{1-V} \sum_{i=1}^{N_p} w_{t-1}^i \left(\hat{\mathbf{p}}_{t|t-1,j}^i - \bar{\mathbf{p}}_{t|t-1,j} \right) \left(\hat{\mathbf{p}}_{t|t-1,j}^i - \bar{\mathbf{p}}_{t|t-1,j} \right)^T \quad (5.50)$$

avec

$$V = \sum_{i=1}^{N_p} (w_{t-1}^i)^2$$

Remarque – Afin de prendre en compte le bruit dynamique, qui est ici supposé gaussien de moyenne nulle et de matrice de covariance Q , alors

- soit il est possible de tirer $\hat{x}_{t|t-1,j}^i \sim p(x_{t,j} | x_{t-1,j}^i)$ puis de calculer (5.50)
- soit la matrice de covariance de l'approximation gaussienne est $\Sigma_{t,j} + Q$ si (5.49) est utilisée.

Les points dont la distance de Mahalanobis (par rapport à l'approximation gaussienne 2D d'une cible) est inférieure au seuil fixé font partie d'une ellipse. Cette ellipse définit un périmètre

dans lequel la cible est estimée être avec une certaine probabilité. Comme énoncé précédemment, aucun *gating* en angle n'est fait. Cependant un *gating* en distance est souhaité: l'ellipse définit donc un espace en distance pour lequel les cellules doivent être sélectionnées. L'ellipse de la cible j est notée $\mathcal{E}_j$ dans la suite.

Soit $r(x, y) = \sqrt{x^2 + y^2}$ la distance cartésienne classique d'un point (x, y) à l'origine (le capteur). Les points de l'ellipse le plus proche et le plus éloigné de l'origine définissent la zone en distance à sélectionner. Ce problème revient à la recherche d'une distance minimum et une distance maximum à définir de la manière suivante, pour chaque cible j ,

$$\begin{aligned} r_{\min,j} &= \min r(x, y) \mid (x, y) \in \mathcal{E}_j \\ r_{\max,j} &= \max r(x, y) \mid (x, y) \in \mathcal{E}_j \end{aligned}$$

En notant $r^c = r(x^c, y^c)$ la distance du centre de la cellule c à l'origine, le *gating* de la cible j donnant la sélection des cellules est défini par

$$G_j = \left\{ c \in \{1, \dots, N_c\} \mid r_{\min,j} - \frac{\Delta_r}{2} \leq r^c \leq r_{\max,j} + \frac{\Delta_r}{2} \right\}$$

avec N_c le nombre de cellules et Δ_r la résolution en distance d'une cellule.

Une illustration de ce principe de *gating* est donnée sur la figure 5.9.

Enfin le *gating* total est l'union des *gatings*

$$G = \bigcup_{j=1}^N G_j$$

Finalement l'image réduite après *gating* est l'union des observations des cellules sélectionnées:

$$z^G = \{z^c \mid c \in G\}$$

Mise en pratique

La recherche du point de l'ellipse le plus éloigné et du point le plus proche du capteur fait appel à l'expression de l'ellipse dans le domaine cartésien, l'expression de la distance du capteur à un point de l'ellipse, et la minimisation et maximisation de l'expression de cette distance.

Pour chaque cible, on retrouve facilement l'expression de l'ellipse grâce l'estimation de la moyenne de la position et de la matrice de covariance. Les expressions qui suivent sont données pour une cible et l'indice j est donc écarté.

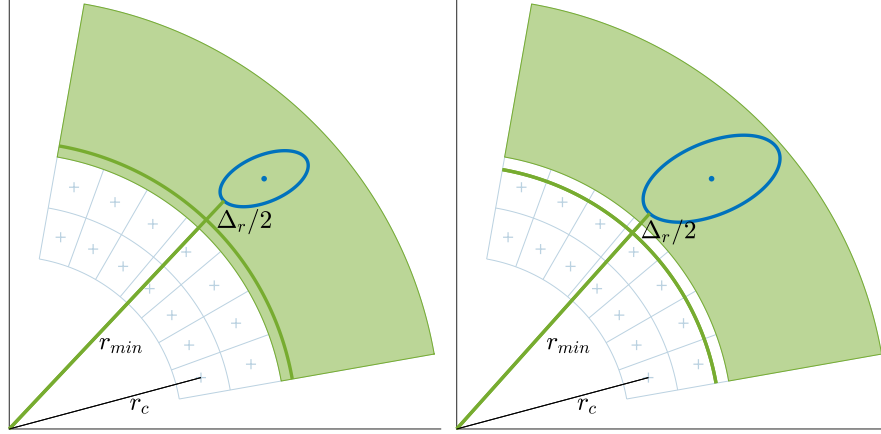


Figure 5.9: Illustration du *gating* – Les cellules sélectionnées pour la fenêtre sont ici représentées en vert. L'illustration donne le résultat pour la recherche de la distance minimale mais il en est de même pour la distance maximale. Les deux situations démontrent bien que chaque cellule par laquelle passe l'ellipse est bien sélectionnée, que le centre de la cellule ait une distance plus petite ou plus grande que la distance minimale.

Le centre de l'ellipse $(x_{\mathcal{E}}, y_{\mathcal{E}})$ correspond à l'estimation de la moyenne de la position de la cible $\bar{\mathbf{p}}_{t|t-1} = (\bar{x}_{t|t-1}, \bar{y}_{t|t-1})$ et donc simplement $x_{\mathcal{E}} = \bar{x}_{t|t-1}$ et $y_{\mathcal{E}} = \bar{y}_{t|t-1}$.

La taille de l'ellipse est donnée à partir du seuil fixé pour la distance de Mahalanobis. Ce seuil s est déterminé à partir de la loi inverse du khi à deux degrés de liberté et du niveau de confiance p souhaité

$$s = \sqrt{-2 \ln(1 - p)}$$

Afin d'avoir accès à l'équation cartésienne de l'ellipse, les paramètres du grand axe a et du petit axe b , et l'angle d'inclinaison θ de l'ellipse doivent être déterminés. Soient λ_1 et λ_2 les valeurs propres de la matrice Σ , et telles que $\lambda_1 \geq \lambda_2$. Soit U le vecteur propre de cette même matrice associé à la valeur propre λ_1 . Alors les paramètres de l'ellipse sont définis par

$$a = s\sqrt{\lambda_1} \quad \text{et} \quad b = s\sqrt{\lambda_2}$$

$$\theta = \arctan\left(\frac{U_y}{U_x}\right)$$

La représentation de l'ellipse et de ses paramètres sont représentés sur la figure 5.10.

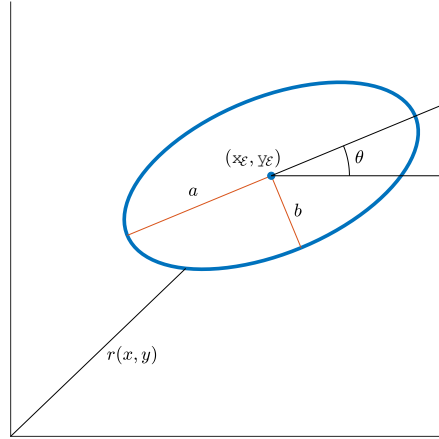


Figure 5.10: Paramètres de l'ellipse

L'équation cartésienne de l'ellipse s'écrit alors

$$\frac{[(x - x_E) \cos \theta + (y - y_E) \sin \theta]^2}{a^2} + \frac{[(x - x_E) \sin \theta - (y - y_E) \cos \theta]^2}{b^2} = 1 \quad (5.51)$$

Quelques manipulations sur l'expression (5.51) ci-dessus permettent de faire clairement apparaître une équation du second degré pour la variable $Y = y - y_E$ (avec des coefficients dépendant de la variable $(x - x_E)$):

$$\left(\frac{\cos^2 \theta}{b^2} + \frac{\sin^2 \theta}{a^2} \right) Y^2 + 2 \left(\frac{1}{a^2} - \frac{1}{b^2} \right) (x - x_E) \cos \theta \sin \theta Y + \left(\frac{\cos^2 \theta}{a^2} + \frac{\sin^2 \theta}{b^2} \right) (x - x_E)^2 - 1 = 0$$

Le discriminant de cette équation s'exprime alors

$$\Delta = 4 \left(\frac{\cos^2 \theta}{b^2} + \frac{\sin^2 \theta}{a^2} - \left(\frac{x - x_E}{ab} \right)^2 \right)$$

En contraignant le discriminant à être positif $\Delta > 0$, afin d'obtenir des solutions réelles à l'équation, le domaine de définition de x pour l'ellipse est déterminé

$$x_E - \sqrt{a^2 \cos^2 \theta + b^2 \sin^2 \theta} < x < x_E + \sqrt{a^2 \cos^2 \theta + b^2 \sin^2 \theta}$$

Les solutions de l'équation, donnant l'expression de y , pour x dans le domaine de définition

de l'ellipse, sont donc

$$y = y_{\mathcal{E}} + \frac{(a^2 - b^2)(x - x_{\mathcal{E}}) \cos \theta \sin \theta \pm ab \sqrt{a^2 \cos^2 \theta + b^2 \sin^2 \theta - (x - x_{\mathcal{E}})^2}}{a^2 \cos^2 \theta + b^2 \sin^2 \theta} \quad (5.52)$$

Cette expression de y définit donc deux arcs représentant l'ellipse complètement. Finalement, l'expression donnant la distance entre n'importe quel point de l'ellipse et le capteur

$$r = r(x) = \sqrt{x^2 + \left(y_{\mathcal{E}} + \frac{d(x - x_{\mathcal{E}}) \pm ab \sqrt{c - (x - x_{\mathcal{E}})^2}}{c} \right)^2} \quad (5.53)$$

avec

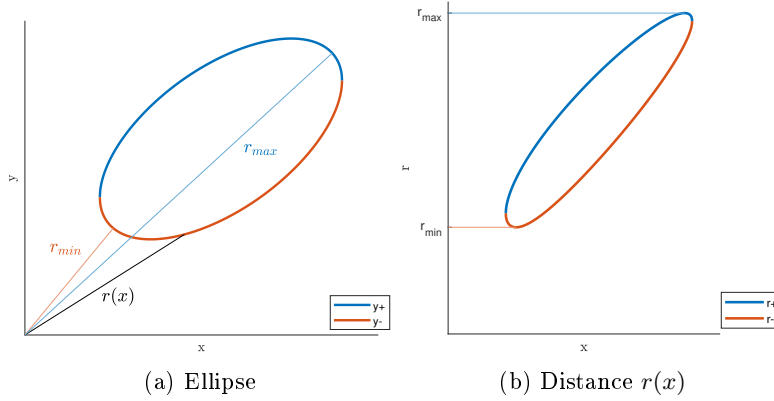
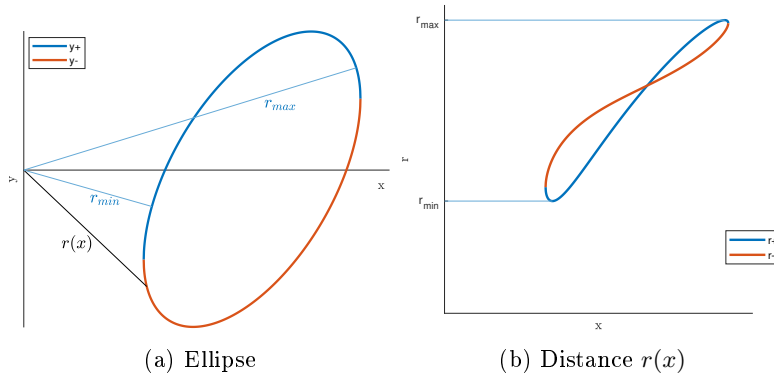
$$c = a^2 \cos^2 \theta + b^2 \sin^2 \theta$$

$$d = (a^2 - b^2) \cos \theta \sin \theta$$

L'expression (5.52) fait clairement apparaître l'expression de deux arcs de l'ellipse (selon si le signe $+$ ou le signe $-$ est choisi dans l'expression). L'expression de la distance (5.53) possède de même deux versions selon le signe choisi dans l'expression. Les versions des expressions (5.52) et (5.53) avec le signe $+$ seront notées y_+ et r_+ , les versions avec le signe $-$ seront notées y_- et r_- . Peut-on déterminer facilement sur quel arc les distances minimum et maximum sont respectivement atteintes ? Cela pourrait simplifier la recherche de ces extrema. Intuitivement, il paraît possible de définir sur quel arc chercher le minimum et le maximum, notamment lorsque l'ellipse ne coupe pas l'axe des abscisses. La figure 5.11 illustre ce cas. Cependant, dans certaines situations, le minimum et le maximum sont atteints sur la même version de la fonction r . Comme illustré sur la figure 5.12, la fonction r atteint son minimum et son maximum sur sa version r_+ . Il est donc difficile dans ce cas d'imaginer une stratégie permettant de savoir sur quelle version de la fonction trouver les extrema.

A partir de cette expression de r il est facile de déterminer la dérivée première et la dérivée seconde par rapport au paramètre x afin de mettre en place des méthodes d'optimisation telles que la descente de gradient ou la méthode de Newton par exemple.

Pour ce problème de minimisation et maximisation, la solution retenue ci-dessus choisit de passer par la représentation cartésienne de l'équation de l'ellipse. Il a été envisagé de faire le même travail sur la représentation paramétrique de l'équation de l'ellipse, mais les expressions obtenues n'apportent pas plus de simplicité au problème. Il a également été envisagé de traiter le problème à l'aide des multiplicateurs de Lagrange, sans gain de simplicité non plus.

Figure 5.11: Ellipse et sa distance associée $r(x)$ - cas 1Figure 5.12: Ellipse et sa distance associée $r(x)$ - cas 2

Extension à l'image radar 3D

Dans le cas où l'image radar comporte également l'information sur la vitesse des objets (radar Doppler), il est envisageable de réaliser le même type de *gating* sur la composante distance de l'image radar, et de garder toutes les cellules de la composante angulaire et de la composante vitesse (ou fréquence Doppler).

Selon la forme de l'impulsion radar, le filtre adapté peut faire apparaître une corrélation entre la distance et la vitesse de la cible (notamment avec un signal de type *chirp*). Si un *gating* est envisagé également en vitesse, cette corrélation pourra être prise en compte.

Remarques et perspectives

Le radar a été conçu et amélioré au fil des années afin d'obtenir une détection précise

et fiable de la cible, dans le but de réaliser un pistage sur données ponctuelles (seuillées). Les "défauts" du radar dans le cadre du pistage classique sur données ponctuelles peuvent devenir en TBD des sources d'information, du moment que le modèle d'observation est bien connu via une bonne détermination de la fonction d'équipement. En effet la présence de lobes secondaires sur la composante angulaire peut venir confirmer la présence d'une cible plutôt que de donner de fausses alarmes venant défier l'algorithme de pistage. Cependant si les lobes sont atténués par des méthodes de réduction, dans le but de diminuer les fausses alarmes pour des données seuillées, de l'information est perdue. Il serait intéressant, dans un cadre TBD, de pouvoir juger de la pertinence des lobes secondaires comme motif pour discriminer les cibles. Il pourrait aussi être envisageable de ne garder que certaines lobes jugés suffisamment significatifs pour réduire d'avantage le nombre de cellules de l'observation.

Chapitre 6

Mesures de performance multicible

Afin de juger et comparer les performances des différents filtres multi-cible, il est nécessaire de définir des mesures de performances évaluant équitablement chacun des filtres. Définir une mesure implique d'établir une référence à laquelle les résultats donnés par les filtres seront comparés. Cette référence devrait être assez générale pour convenir à chacun des filtres, et correspond généralement aux données réelles ou au cas le plus favorable.

Les deux sections de ce chapitre présentent deux mesures différentes qui permettent d'évaluer différemment les performances de filtrage multi-objet.

6.1 Mesure par Schéma d'Affectation Optimale (Optimal Subpattern Assignment (OSPA) Metric)

Comme décrit dans [SVV08], afin d'introduire et d'adapter la notion d'erreur et d'écart en distance nécessaires à l'évaluation des performances des filtres multi-objets, les auteurs ont développé une mesure intuitive et adéquate. L'idée est de donner simultanément une erreur sur la cardinalité et l'état des cibles, entre l'estimation des cibles extraite du filtrage et une population de référence pour les cibles. Cette mesure dépend de deux paramètres, $p \geq 1$ et $c > 0$, ajustables selon leur interprétation. La référence est la vérité terrain (données réelles), c'est-à-dire l'ensemble des états réels des cibles.

Soit

$$d^{(c)}(x, y) = \min(c, d(x, y))$$

la distance entre $x, y \in \mathcal{W}$ (fenêtre d'observation) qui est coupée à $c > 0$, et Π_j l'ensemble des permutations sur $\{1, 2, \dots, j\}$ pour tout $j \in \mathbb{N}_+$. Pour $1 \leq p < \infty$, $c > 0$, et sous-ensembles arbitraires $X = \{x_1, \dots, x_m\}$ et $Y = \{y_1, \dots, y_n\}$ de $\mathcal{W}$, où $m, n \in \mathbb{N}$, est défini

$$\bar{d}_p^{(c)}(X, Y) = \begin{cases} 0, & m = n = 0 \\ \left(\frac{1}{n} \left(\min_{\pi \in \Pi_n} \sum_{i=1}^m d^{(c)}(x_i, y_{\pi(i)})^p + c^p(n-m) \right) \right)^{1/p}, & m \leq n \\ \bar{d}_p^{(c)}(Y, X), & m > n \end{cases}$$

La construction de la distance OSPA est assez intuitive:

- L'association des cibles de l'ensemble le plus petit à celles de l'ensemble le plus grand est établie afin de minimiser la distance totale, en considérant comme erreur de cardinalité les associations trop lointaines (distance supérieure à c) : $\min_{\pi \in \Pi_n} \sum_{i=1}^m d^{(c)}(x_i, y_{\pi(i)})^p$
- La différence de cardinalité entre les deux ensembles est pénalisée : $c^p(n-m)$

Ce qui permet de donner une interprétation pour chacun des paramètres afin de les choisir selon l'utilisation voulue:

- Le paramètre d'ordre p détermine la sensibilité de la mesure en pénalisant les estimations aberrantes (outliers).
- Le paramètre de coupure c détermine à quel point la mesure pénalise les erreurs de cardinalité par opposition aux erreurs de localisation.

La façon dont les paramètres sont ajustés permet de voir simultanément ou séparément les erreurs d'estimation d'état ou d'estimation de cardinalité. Pour cette analyse, le paramètre d'ordre est fixé à $p = 2$, tandis que le paramètre de coupure est fixé à $c = 10$. Ce choix pénalise les outliers et l'erreur en cardinalité peut facilement être observée lors de l'affichage de la distance OSPA dans le temps.

Cependant, la distance OSPA compare seulement les états estimés aux données réelles. L'estimation des états extraite dépend de la technique d'estimation choisie. Or les filtres donnent des densités qui sont plus informatives, il est donc concevable de vouloir comparer leurs performances sur les distributions issues du filtrage. Il est alors nécessaire de définir une nouvelle mesure pour comparer les distributions estimées données en sortie des différents filtres.

6.2 Divergence de Kullback-Leibler

La divergence de Kullback-Leibler (KL), définie par Kullback dans [Kul68], est une mesure non symétrique de dissimilarité entre deux distributions de probabilité. Elle représente donc la distance entre deux modèles (les distributions) considérés, ce qui permet de les comparer. La

divergence de KL définie pour des distributions de probabilité de variables aléatoires continues est

$$D_{KL}(f\|g) = \int f(x) \log \frac{f(x)}{g(x)} dx$$

où f et g sont les densités des distributions de probabilité à comparer, avec par convention $\log \frac{f(x)}{g(x)} = 0$ si $f(x) = g(x) = 0$ et $\log \frac{f(x)}{g(x)} = \infty$ si $f(x) > 0$ et $g(x) = 0$.

La divergence de KL est étroitement liée à la théorie de l'information, et se définit comme étant "l'information perdue lorsque g est utilisé pour approximer f ". Dans le but de comparer les performances de deux filtres différents, leurs divergences de KL par rapport à une distribution de référence seront comparées. Typiquement f représente une connaissance exacte du système, la distribution des données réelles, ou un cas très favorable, par exemple lorsque l'association exacte de données est connue. Tandis que g représente typiquement la distribution issue du modèle, du filtre choisis. Si g_1 (resp. g_2) est l'estimation donnée par un filtre 1 (resp. filtre 2), alors si $D_{KL}(f\|g_1) \geq D_{KL}(f\|g_2)$, cela signifie que g_1 est plus proche, ou perd moins d'information, que g_2 par rapport à f , donc ses performances sont meilleures.

Conclusion et perspectives

Pour répondre au défi que représente la poursuite multi-cibles en contexte [TBD](#), notamment en traitant une image-radar, les filtres particuliers offrent une solution permettant de tenir compte du modèle multi-cible et des fortes non linéarités apportées par le modèle d'observation. Au sein de la famille des filtres particuliers sont présentes de nombreuses méthodes et variantes toujours en développement pour répondre à des problèmes de plus en plus complexes, et qu'il est bon d'étudier et de comparer pour un problème donné. Ainsi les filtres particuliers utilisant les méthodes de Monte Carlo par chaînes de Markov sont assez robustes lorsque la chaîne permet une bonne exploration de l'espace d'état. Le filtre particulier [IPMCMC](#) illustre ces méthodes de Monte Carlo par chaînes de Markov en choisissant un noyau de transition adéquat au problème multi-cible, parallélisant les calculs et introduisant du *crossover* via la sélection du nouvel état dans la chaîne de Markov. Le filtre particulier auxiliaire permet d'intégrer l'observation en amont de la sélection des particules à propager, ce qui est intéressant en [TBD](#) compte tenu de l'information précieuse contenue dans le signal brut. Une version complète et adaptée au cadre multi-cibles est illustrée à travers le filtre particulier [ATRAPP](#). L'utilisation en amont de l'observation est cependant assez coûteuse en temps de calcul. Sur un scénario où les cibles restent plutôt éloignées les unes des autres, le choix se porte sur le filtre [IPMCMC](#) qui obtient de meilleurs résultats dans un temps bien plus court. Lorsque le scénario voit des cibles se rapprocher, le coût de calcul du filtre [ATRAPP](#) se voit justifié par l'obtention de meilleurs résultats d'estimation de l'état des cibles.

Afin de tirer profit des observations brutes pour l'échantillonnage, à la manière du filtre particulier auxiliaire, mais sans avoir recours à l'hypothèse d'indépendance postérieure des cibles, une approche originale a été proposée grâce à l'introduction des transitions markoviennes auxiliaires. Deux types de transitions sont proposées: la matrice de transition auxiliaire, qui à un indice (de particule) fait correspondre un autre indice avec une certaine probabilité, et la densité de transition auxiliaire, qui à un indice (de particule) fait correspondre un état selon une densité de probabilité. Appliquées au cas multi-cibles, ces transitions permettent

de générer des particules brassées, réalisant ainsi du *crossover* et apportant de la diversité dans les particules. Les transitions sont donc des paramètres de conception, définis dans ce manuscrit à l'aide du critère d'optimalité du minimum de variance des poids. La variance des poids obtenue avec la densité de transition auxiliaire optimale est réduite par rapport à la variance des poids obtenue avec la matrice de transition auxiliaire optimale. Cette approche est une généralisation du filtre particulaire auxiliaire qui en est un cas particulier (grâce au choix des paramètres de conception).

Le modèle d'observation a été développé et propose un modèle général d'une image-radar dont le signal reçu est complexe avec un retour des cibles d'amplitudes fluctuantes (selon les modèles de Swerling). Selon le modèle de Swerling considéré, si le traitement est effectué sur le signal complexe ou bien sur son module au carré, les vraisemblances à intégrer aux algorithmes sont données. Une approche originale de *gating*, permettant de réduire la dimensions de l'observation, est développée, en s'appuyant sur une approximation gaussienne de la distribution multi-cibles, et en s'appuyant sur la forme de la fonction d'équipement radar qui présente notamment des lobes secondaires.

Le domaine de la poursuite multi-cibles est vaste et de nombreuses pistes de recherche peuvent être poursuivies:

Nombre inconnu et variable de cibles

Les cibles apparaissent et disparaissent dans le champ de vision du ou des capteurs et leur nombre est inconnu. Il est donc nécessaire de mettre en œuvre une gestion de l'apparition et de la disparition du nombre de cibles, et d'existence des cibles afin d'obtenir une estimation de leur nombre. Le filtre particulaire [IPMCMC](#) possède une extension permettant de gérer un nombre inconnu et variable de cibles. Il s'agit du filtre particulaire [Interacting Population-based Reversible Jump Markov Chain Monte Carlo \(IPRJMC\)](#) qui gère l'augmentation (ou diminution) du nombre de cibles (et donc de l'espace d'état) via les sauts réversibles dans la chaîne de Markov. Une méthode permettant également de gérer cette problématique est l'ajout d'une variable d'existence pour chacune des cibles, ainsi que de probabilités de survies. Le cadre des *Random Finite Sets* propose également une solution afin de gérer un nombre variable d'éléments. L'apparition potentielle de cibles implique également la mise en place d'une solution d'initialisation des pistes. Leur disparition peut être gérée par une probabilité d'existence passant sous un seuil donné ou une sortie du champ de vision.

Système d'initialisation des cibles

La création d'un système efficace d'initialisation des cibles est indispensable au bon fonctionnement opérationnel d'algorithmes de poursuite multi-cibles. Dans le cas de données seuillées ponctuelles, l'initialisation est déclenchée par la présence d'un point détecté (cible ou fausse alarme). Dans un contexte [TBD](#), il serait vain de faire appel à une détection pour initialiser les cibles, puisqu'il s'agit de poursuivre des mobiles qui ont peu de chance d'être détectés. Dans les simulations, il est aisé de proposer une initialisation autour du point de départ connu (car simulé) de la cible au moment où elle apparaît, et vérifier a minima que l'algorithme réussit à suivre la cible. Sans cette information sur le départ d'une cible, il est possible d'imaginer une initialisation uniforme de particules sur l'espace d'état: cela semble cependant très coûteux. En considérant le signal radar reçu tel une image, et connaissant la forme de la fonction d'équipement radar, une détection de motifs dans l'image pourrait être envisagée, ou bien une analyse sur plusieurs images simultanément, à la recherche d'un début de trajectoire afin de pouvoir initialiser une cible.

Estimation du paramètre d'intensité de la cible

Le paramètre d'intensité de la cible intervient dans le modèle d'observation et est propre à chacune des cibles selon sa capacité à réfléchir l'onde radar. Il fait donc partie des paramètres inconnus pouvant être estimés. Le [RSB](#) de la cible est directement lié au paramètre d'intensité (et à la distance cible-capteur). Si l'amplitude du signal renvoyé par la cible est considéré comme fluctuant au cours du temps, le paramètre d'intensité lui ne l'est pas et est considéré comme constant. En réalité il peut varier, selon par exemple l'orientation de la cible et sa face présentée au capteur. Son estimation au fur et à mesure peut capter ses variations au cours du temps. Une méthode par exemple serait d'inclure ce paramètre à l'espace d'état des cibles et de lui donner une dynamique artificielle légère permettant d'explorer différentes valeurs lors du filtrage particulière.

Estimation du modèle de Swerling

Le modèle de Swerling des cibles n'est pas non plus connu en réalité. Savoir quel modèle de Swerling régit l'amplitude du signal retour de chacune des cibles pourrait être une donnée intéressante afin d'utiliser la fonction de vraisemblance adéquate à la situation. Savoir qu'une cible fluctue avec un modèle de Swerling 3 justifie de recourir à des calculs plus compliqués, mais savoir qu'une cible fluctue avec un modèle de Swerling 0 permet de simplifier les calculs. Si plusieurs cibles sont présentes dans la scène mais ne fluctuent pas selon le même modèle de Swerling, il faudrait envisager des fonctions de vraisemblances hybrides, à déterminer donc

selon les méthodes proposées dans la section 5.3.

Simulations supplémentaires

Des simulations supplémentaires pourraient être menées notamment pour:

- comparer l'apport du traitement sur le signal radar complexe plutôt que sur le signal réel,
- comparer un traitement en TBD avec un traitement avec des données ponctuelles, sur des cibles à faible RSB, mais également à fort RSB pour s'assurer que la méthode TBD fonctionne aussi pour de telles cibles. Dans ce cas il faudrait développer un détecteur et extracteur de données ponctuelles pour obtenir les points à partir du signal brut,
- comparer les méthodes présentées dans ce manuscrit à des méthodes *batch* (brièvement présentées au paragraphe 2.2.1).

Simulations du filtre particulaire avec transition markovienne auxiliaire

Le développement théorique de cet algorithme original a été mené dans ce manuscrit. Afin d'appuyer son intérêt, des simulations numériques pourront être menées afin de le comparer dans un premier temps avec un filtre particulaire auxiliaire classique (dont il est une généralisation), puis dans un deuxième temps le comparer avec des algorithmes dans le cas multi-cibles, afin de mesurer l'apport du *crossover* géré avec les transitions auxiliaires.

Données réelles

L'acquisition de données réelles pour réaliser la poursuite multi-cibles en contexte TBD serait un vrai apport. Les relevés existants sont bien souvent effectués pour réaliser la détection et obtenir des données ponctuelles, mais non pour réaliser la poursuite en contexte TBD. Ainsi les paramètres nécessaires au modèle d'observation ne sont pas toujours collectés et disponibles dans les jeux de données. Aussi le radar est choisi pour avoir de bonnes performances de détection, c'est-à-dire une bonne résolution et l'atténuation des lobes secondaires par exemple. Le signal complexe est peu souvent enregistré alors qu'il peut apporter une information supplémentaire lors d'un traitement sur signal brut. Les lobes secondaires, plutôt qu'être atténués, pourraient également contenir de l'information pertinente pour un traitement en TBD.

Références

- [AM79] Anderson, B. D. O., et Moore, J. B. *Optimal filtering*. Prentice-Hall, Englewood Cliffs, New Jersey, 1979. [9](#)
- [AMGC02] Arulampalam, S., Maskell, S., Gordon, N., et Clapp, T. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188, 2002. [19](#)
- [Bar85] Barniv, Y. Dynamic programming solution for detecting dim moving targets. *IEEE Transactions on Aerospace and Electronic Systems*, AES-21(1):144–156, 1985. [55](#)
- [BBS88] Blom, H. A. P., et Bar-Shalom, Y. The interacting multiple model algorithm for systems with Markovian switching coefficients. *IEEE Transactions on Automatic Control*, 33(8):780–783, 1988. [47](#)
- [BD01] Boers, Y., et Driessen, H. Particle filter based detection for tracking. In *Proceedings of the 2001 American Control Conference. (Cat. No.01CH37148)*, volume 6, pages 4393–4397 vol.6, 2001. [55](#)
- [BDB12] Bocquel, M., Driessen, H., et Bagchi, A. Multitarget tracking with interacting population-based MCMC-PF. In *15th International Conference on Information Fusion (FUSION 2012)*, pages 74–81, 2012. [7](#), [33](#), [60](#), [61](#)
- [BDB13] Bocquel, M., Driessen, H., et Bagchi, A. Multitarget tracking with IP reversible jump MCMC-PF. In *Proceedings of the 16th International Conference on Information Fusion*, pages 556–563, 2013. [61](#), [66](#)
- [BDV⁺03] Boers, Y., Driessen, J. N., Verschure, F., Heemels, W. P. M. H., et Juloski, A. A multi target track before detect application. In *2003 Conference on Computer Vision and Pattern Recognition Workshop*, volume 9, pages 104–104, 2003. [122](#)

- [Bla86] Blackman, S. S. *Multiple-target Tracking with Radar Applications*. Radar Library. Artech House, 1986. [46](#), [48](#)
- [Boc13] Bocquel, M. *Random Finite Sets in Multi-Targets Tracking Efficient Sequential MCMC implementation*. PhD thesis, University of Twente, 2013. [141](#)
- [BSF88] Bar-Shalom, Y., et Fortmann, T. E. *Tracking and Data Association*. Mathematics in Science and Engineering Series. Academic Press, 1988. [46](#), [47](#), [48](#)
- [CCF99] Carpenter, J., Clifford, P., et Fearnhead, P. Improved particle filter for nonlinear problems. *IEE Proceedings - Radar, Sonar and Navigation*, 146(1):2–7, 1999. [25](#)
- [CEW94] Carlson, B. D., Evans, E. D., et Wilson, S. L. Search radar detection and track with the Hough transform. i. system concept. *IEEE Transactions on Aerospace and Electronic Systems*, 30(1):102–108, 1994. [55](#)
- [CG92] Casella, G., et George, E. I. Explaining the Gibbs sampler. *American Statistician*, 43(3):167–174, 1992. [37](#)
- [CG95] Chib, S., et Greenberg, E. Understanding the Metropolis-Hastings algorithm. *The American Statistician*, 49(4):327–335, 1995. [37](#)
- [CLG19] Cuillery, A., et Le Gland, F. Track-before-detect on radar image observation with an adaptive auxiliary particle filter. In *2019 22th International Conference on Information Fusion (FUSION)*. IEEE, 2019.
- [CLG21a] Cuillery, A., et Le Gland, F. Particle filters with auxiliary markov transition. application to crossover and to multitarget tracking. In *2021 24th International Conference on Information Fusion (FUSION)*. IEEE, 2021.
- [CLG21b] Cuillery, A., et Le Gland, F. Particle filters with auxiliary Markov transition. Application to crossover and to multitarget tracking. Publication Hal hal-03467603, 2021.
- [CMR05] Cappé, O., Moulines, E., et Ryden, T. *Inference in Hidden Markov Models*. Springer-Verlag New York, 2005. [20](#)
- [DCM05] Douc, R., Cappé, O., et Moulines, E. Comparison of resampling schemes for particle filtering. In *ISPA 2005. Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis*, pages 64–69, 2005. [21](#)

-
- [DdMR00] Doucet, A., de Freitas, N., Murphy, K. P., et Russell, S. J. Rao-blackwellised particle filtering for dynamic bayesian networks. In *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence*, UAI '00, pages 176–183, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc. [33](#)
 - [DGA00] Doucet, A., Godsill, S., et Andrieu, C. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, 10(3):197–208, 2000. [29](#)
 - [DJ11] Doucet, A., et Johansen, A. M. A tutorial on particle filtering and smoothing: Fifteen years later. In Crisan, D., et Rozovsky, B., editors, *Handbook of Nonlinear Filtering*. Oxford University Press, 2011. [19](#)
 - [DLR77] Dempster, A. P., Laird, N. M., et Rubin, D. B. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977. [48](#)
 - [DMO09] Douc, R., Moulines, E., et Olsson, J. Optimality of the auxiliary particle filter. *Probability and Mathematical Statistics*, 29(1):1–28, 2009. [24](#), [25](#), [29](#)
 - [DRC12] Davey, S. J., Rutten, M. G., et Cheung, B. Using phase to improve track-before-detect. *IEEE Transactions on Aerospace and Electronic Systems*, 48(1):832–849, 2012. [122](#), [123](#), [129](#)
 - [DVJ03] Daley, D. J., et Vere-Jones, D. *An introduction to the theory of point processes. Vol. I.* Probability and its Applications. Springer, New York, second edition, 2003. Elementary theory and methods. [49](#)
 - [DVJ08] Daley, D. J., et Vere-Jones, D. *An introduction to the theory of point processes. Vol. II.* Probability and its Applications. Springer, New York, second edition, 2008. General theory and structure. [49](#)
 - [EMBD18] Elvira, V., Martino, L., Bugallo, M. F., et Djurić, P. M. In search for improved auxiliary particle filters. In *2018 26th European Signal Processing Conference (EUSIPCO)*, pages 1637–1641. IEEE, 2018. [33](#)
 - [FG10] Fallon, M. F., et Godsill, S. Acoustic source localization and tracking using track before detect. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(6):1228–1242, 2010. [58](#)
 - [FG12] Fallon, M. F., et Godsill, S. J. Acoustic source localization and tracking of a time-varying number of speakers. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(4):1409–1415, 2012. [58](#)

- [FSK10] Fritsche, C., Schön, T., et Klein, A. The marginalized auxiliary particle filter. In *2009 3rd IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, pages 289 – 292, 2010. [33](#)
- [GGB⁺02] Gustafsson, F., Gunnarsson, F., Bergman, N., Forssell, U., Jansson, J., Karlsson, R., et Nordlund, P.-J. Particle filters for positioning, navigation, and tracking. *IEEE Transactions on signal processing*, 50(2):425–437, 2002. [33](#)
- [GMN97] Goodman, I. R., Mahler, R. P., et Nguyen, H. T. *Mathematics of Data Fusion*. Kluwer Academic Publishers, Norwell, MA, USA, 1997. [49](#)
- [Goo63] Goodman, N. R. Statistical analysis based on a certain multivariate complex Gaussian distribution (an introduction). *Ann. Math. Statist.*, 34(1):152–177, 1963. [105](#)
- [Gor97] Gordon, N. A hybrid bootstrap filter for target tracking in clutter. *IEEE Transactions on Aerospace and Electronic Systems*, 33(1):353–358, 1997. [49](#)
- [GR07] Gradshteyn, I. S., et Ryzhik, I. M. *Tables of integrals, series, and products*. Elsevier, seventh edition, 2007. [132](#)
- [Gre95] Green, P. J. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82(4):711–732, 1995. [61](#), [66](#)
- [GSS93] Gordon, N. J., Salmond, D. J., et Smith, A. F. M. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEEE Proceedings - Radar and Signal Processing*, 140(2):107–113, 1993. [19](#), [29](#)
- [HBS89] Houles, A., et Bar-Shalom, Y. Multisensor tracking of a maneuvering target in clutter. *IEEE Transactions on Aerospace and Electronic Systems*, 25(2):176–189, 1989. [47](#)
- [HLCP02a] Hue, C., Le Cadre, J.-P., et Pérez, P. Sequential Monte Carlo methods for multiple target tracking and data fusion. *IEEE Trans. on Signal Processing*, 50(2):309–325, 2002. [49](#)
- [HLCP02b] Hue, C., Le Cadre, J.-P., et Pérez, P. Tracking multiple objects with particle filtering. *IEEE Transactions on Aerospace and Electronic Systems*, 38(3):791–812, 2002. [49](#)
- [IBD14] Iglesias García, F. J., Bocquel, M., et Driessen, H. Advanced IP-MCMC-PF design ingredients. In *17th International Conference on Information Fusion (FUSION 2014)*, pages 1–8, 2014. [61](#)

-
- [IBMD15] Iglesias García, F. J., Bocquel, M., Mandal, P. K., et Driessen, H. Acceptance probability of IP-MCMC-PF: revisited. In *2015 Sensor Data Fusion: Trends, Solutions, Applications (SDF)*, pages 1–6, 2015. [61](#)
 - [JD08] Johansen, A. M., et Doucet, A. A note on auxiliary particle filters. *Statistics and Probability Letters*, pages 1498–1504, 2008. [24](#), [25](#)
 - [KBB16] Kreucher, C., Bierdz, P., et Bell, K. Optimal exploitation of fluctuating target measurements. In *2016 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pages 1088–1091, 2016. [122](#)
 - [KBD05] Khan, Z., Balch, T., et Dellaert, F. MCMC-based particle filtering for tracking a variable number of interacting targets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(11):1805–1819, 2005. [33](#), [61](#), [66](#)
 - [Kon92] Kong, A. A note on importance sampling using standardized weights. Technical Report 348, University of Chicago, Department of Statistics, 1992. [18](#)
 - [Kul68] Kullback, S. Information theory and statistics, 1968. [151](#)
 - [LC02] Le Chevalier, F. *Principles of Radar and Sonar Signal Processing*. Artech House, 2002. [99](#), [113](#)
 - [Lep16] Lepoutre, A. *Detection and tracking in Track-Before-Detect context using particle filtering*. Theses, Université de Rennes 1, 2016. [141](#)
 - [Liu01] Liu, J. S. *Monte Carlo strategies in Scientific Computing*. Springer, 2001. [21](#)
 - [LRL13] Lepoutre, A., Rabaste, O., et Le Gland, F. Exploiting amplitude spatial coherence for multi-target particle filter in track-before-detect. In *Proceedings of the 16th International Conference on Information Fusion*, pages 319–326, 2013. [122](#)
 - [LRL16] Lepoutre, A., Rabaste, O., et Le Gland, F. Multitarget likelihood computation for track-before-detect applications with amplitude fluctuations of type Swerling 0, 1 and 3. *IEEE Transactions on Aerospace and Electronic Systems*, 52(3):1089–1107, 2016. [122](#), [123](#), [130](#), [137](#)
 - [Mah03] Mahler, R. P. S. Multitarget Bayes filtering via first-order multitarget moments. *IEEE Transactions on Aerospace and Electronic Systems*, 39(4):1152–1178, 2003. [49](#), [52](#)
 - [Mah07a] Mahler, R. PHD filters of higher order in target number. *IEEE Transactions on Aerospace and Electronic Systems*, 43(4):1523–1543, 2007. [49](#), [52](#)

- [Mah07b] Mahler, R. P. S. *Statistical multisource-multitarget information fusion*. Artech House Inc., Norwood, MA, USA, 2007. [53](#)
- [MB08a] McDonald, M., et Balaji, B. Impact of measurement model mismatch on nonlinear track-before-detect performance. In *2008 IEEE Radar Conference*, pages 1–6, 2008. [58](#)
- [MB08b] McDonald, M., et Balaji, B. Track-before-detect using Swerling 0, 1, and 3 target models for small manoeuvring maritime targets. *EURASIP Journal on Advances in Signal Processing*, 2008(1):326259, 2008. [56](#), [58](#), [122](#), [123](#)
- [MB11] McDonald, M., et Balaji, B. Impact of measurement model mismatch on nonlinear track-before-detect performance for maritime radar surveillance. *IEEE Journal of Oceanic Engineering*, 36(4):602–614, 2011. [59](#)
- [ME04] Musicki, D., et Evans, R. Joint integrated probabilistic data association: JIPDA. *IEEE Transactions on Aerospace and Electronic Systems*, 40(3):1093–1099, 2004. [48](#)
- [MEF12] Mahler, R., et El-Fallah, A. An approximate CPHD filter for superpositional sensors. In *Proceedings of the SPIE*, volume 8392, 2012. [56](#)
- [MES94] Musicki, D., Evans, R., et Stankovic, S. Integrated probabilistic data association. *IEEE Transactions on Automatic Control*, 39(6):1237–1241, 1994. [47](#)
- [MM04] Mušicki, D., et Morelande, M. Gate volume estimation for target tracking. In *7th International Conference on Information Fusion, Fusion 2004*, 2004. [139](#), [140](#), [141](#)
- [NCM13] Nannuru, S., Coates, M., et Mahler, R. Computationally-tractable approximate PHD and CPHD filters for superpositional sensors. *IEEE Journal of Selected Topics in Signal Processing*, 7(3):410–420, 2013. [30](#), [56](#)
- [NS97] Noyer, J.-C., et Salut, G. Filtrage particulière du signal radar brut sur cible ponctuelle. *Traitement du Signal*, 14(1):43–61, 1997. [55](#), [106](#)
- [PK15] Papi, F., et Kim, D. Y. A particle multi-target tracker for superpositional measurements using labeled random finite sets. *IEEE Transactions on Signal Processing*, 63(16):4348–4358, 2015. [56](#)
- [PS99] Pitt, M. K., et Shephard, N. Filtering via simulation: Auxiliary particle filters. *Journal of the American Statistical Association*, 94(446):590–599, 1999. [24](#), [27](#), [29](#)

-
- [RAG04] Ristic, B., Arulampalam, S., et Gordon, N. *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Artech House, Boston, London, 2004. [19](#)
 - [RC04] Robert, C. P., et Casella, G. *Monte Carlo statistical methods*. Springer texts in statistics. Springer, New York, Berlin, Heidelberg, second edition edition, 2004. Reprint. 1999, 2002. [37](#), [38](#), [39](#), [40](#)
 - [Rei79] Reid, D. An algorithm for tracking multiple targets. *IEEE Transactions on Automatic Control*, 24(6):843–854, 1979. [48](#)
 - [Reu14] Reuter, S. *Multi-object tracking using random finite sets*. PhD thesis, Ulm University, 2014. [54](#)
 - [RGM04] Rutten, M. G., Gordon, N. J., et Maskell, S. Efficient particle-based track-before-detect in Rayleigh noise. In *7th International Conference on Information Fusion (FUSION 2004)*, 2004. [56](#)
 - [RGM05] Rutten, M. G., Gordon, N. J., et Maskell, S. Recursive track-before-detect with target amplitude fluctuations. *IEEE Proceedings - Radar, Sonar and Navigation*, 152(5):345–352, 2005. [56](#)
 - [Rob15] Robert, C. P. The Metropolis-Hastings algorithm. *ArXiv e-prints*, 2015. [37](#)
 - [RWS95] Rago, C., Willett, P., et Streit, R. A comparison of the JPDAF and PMHT tracking algorithms. In *1995 International Conference on Acoustics, Speech, and Signal Processing*, volume 5, pages 3571–3574 vol.5, 1995. [48](#)
 - [Sĩ3] Särkkä, S. *Bayesian Filtering and Smoothing*. Cambridge University Press, 2013. [9](#)
 - [SB01] Salmond, D. J., et Birch, H. A particle filter for track-before-detect. In *Proceedings of the 2001 American Control Conference. (Cat. No.01CH37148)*, volume 5, pages 3755–3760 vol.5, 2001. [55](#)
 - [SCSC16] Saucan, A.-A., Chonavel, T., Sintès, C., et Caillec, J.-M. L. CPHD-DOA tracking of multiple extended sonar targets in impulsive environments. *IEEE Transactions on Signal Processing*, 64(5):1147–1160, 2016. [30](#), [57](#)
 - [SDHC16] Schlangen, I., Delande, E., Houssineau, J., et Clark, D. E. A phd filter with negative binomial clutter. In *2016 19th International Conference on Information Fusion (FUSION)*, pages 658–665, 2016. [53](#)

- [SDHC18] Schlangen, I., Delande, E. D., Houssineau, J., et Clark, D. E. A second-order phd filter with mean and variance in target number. *IEEE Transactions on Signal Processing*, 66(1):48–63, 2018. [53](#)
- [SGN05] Schön, T., Gustafsson, F., et Nordlund, P.-J. Marginalized particle filters for mixed linear/nonlinear state-space models. *IEEE Transactions on signal processing*, 53(7):2279–2289, 2005. [33](#), [35](#), [36](#)
- [Sko62] Skolnik, M. I. *Introduction to Radar Systems*. McGraw-Hill, 1962. [99](#), [116](#)
- [SL94] Streit, R. L., et Luginbuhl, T. E. Maximum likelihood method for probabilistic multihypothesis tracking. *Proceedings of SPIE - The International Society for Optical Engineering*, 2235:2235 – 2235 – 12, 1994. [48](#)
- [SPGC09] Septier, F., Pang, S. K., Godsill, S., et Carmi, A. Tracking of coordinated groups using marginalised mcmc-based particle algorithm. In *2009 IEEE Aerospace conference*, pages 1–11, 2009. [33](#)
- [SVV08] Schuhmacher, D., Vo, B.-T., et Vo, B.-N. A Consistent Metric for Performance Evaluation of Multi-Object Filters. *IEEE Transactions on Signal Processing*, 56(8):3447–3457, 2008. [150](#)
- [Swe60] Swerling, P. Probability of detection for fluctuating targets. *IRE Transactions on Information Theory*, 6(2):269–308, 1960. [116](#)
- [TBS98] Tonissen, S. M., et Bar-Shalom, Y. Maximum likelihood track-before-detect with fluctuating target amplitude. *IEEE Transactions on Aerospace and Electronic Systems*, 34(3):796–809, 1998. [55](#)
- [UMGFG17] Úbeda Medina, L., García-Fernández, A. F., et Grajal, J. Adaptive auxiliary particle filter for track-before-detect with multiple targets. *IEEE Transactions on Aerospace and Electronic Systems*, 53(5):2317–2330, 2017. [7](#), [56](#), [57](#), [67](#)
- [VM06] Vo, B.-N., et Ma, W.-K. The Gaussian mixture probability hypothesis density filter. *IEEE Transactions on Signal Processing*, 54(11):4091–4104, 2006. [53](#)
- [Vo08] Vo, B.-T. *Random finite sets in Multi-object filtering*. PhD thesis, The University of Western Australia, 2008. [50](#), [53](#)
- [VSD05] Vo, B.-N., Singh, S., et Doucet, A. Sequential Monte Carlo methods for multi-target filtering with random finite sets. *IEEE Transactions on Aerospace and Electronic Systems*, 41(4):1224–1245, 2005. [53](#)

-
- [VV13] Vo, B.-T., et Vo, B.-N. Labeled random finite sets and multi-object conjugate priors. *IEEE Transactions on Signal Processing*, 61(13):3460–3475, 2013. [54](#)
- [VVC07] Vo, B.-T., Vo, B.-N., et Cantoni, A. Analytic implementations of the cardinalized probability hypothesis density filter. *IEEE Transactions on Signal Processing*, 55:3553–3567, 2007. [53](#)
- [VVC09] Vo, B.-T., Vo, B.-N., et Cantoni, A. The cardinality balanced multi-target multi-bernoulli filter and its implementations. *IEEE Transactions on Signal Processing*, 57(2):409–423, 2009. [53](#)
- [VVPS10] Vo, B.-N., Vo, B.-T., Pham, N.-T., et Suter, D. Joint detection and estimation of multiple objects from image observations. *IEEE Transactions on Signal Processing*, 58(10):5129–5141, 2010. [55](#)
- [YMKY13] Yi, W., Morelande, M. R., Kong, L., et Yang, J. A computationally efficient particle filter for multitarget tracking using an independence approximation. *IEEE Transactions on Signal Processing*, 61(4):843–856, 2013. [68](#)

Annexe A

Particle filters with auxiliary Markov transition

Cette annexe est une pré-publication venant compléter l'étude des transitions markoviennes auxiliaires développées en section [3.3](#).



Particle filters with auxiliary Markov transition. Application to crossover and to multitarget tracking

Audrey Cuillery, François Le Gland

► To cite this version:

Audrey Cuillery, François Le Gland. Particle filters with auxiliary Markov transition. Application to crossover and to multitarget tracking. 2021. hal-03467603

HAL Id: hal-03467603

<https://hal.inria.fr/hal-03467603>

Preprint submitted on 6 Dec 2021

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.

Particle filters with auxiliary Markov transition

Application to crossover and to multitarget tracking

Audrey Cuillery
université de Rennes 1
campus de Beaulieu
35042 Rennes Cédex, France
email: `acuillery@gmail.com`

François Le Gland
INRIA Rennes and IRMAR
campus de Beaulieu
35042 Rennes Cédex, France
email: `francois.le_gland@inria.fr`

Abstract

This work introduces a new class of particle filters, that include an auxiliary Markov transition in their design. Actually, it was motivated by potential application to multitarget tracking, but the solution provided may be of practical interest elsewhere.

This work also shows how to include a crossover step in sequential Monte Carlo methods. It is well known that sequential Monte Carlo methods can be interpreted in terms of implementing *selection* and *mutation* steps, using the language of evolutionary algorithms. However, most general evolutionary algorithms also include a *crossover* step that has not been considered so far in sequential Monte Carlo methods.

A prototypical situation where crossover is needed, or at least could be useful, is multitarget tracking. In multitarget tracking, it may happen that some targets in a multitarget particle are good proxies, but are not going to be selected just because the other targets in the same multitarget particle are bad proxies. This is unfair, and a better design would be to produce shuffled multitarget particles such that the particle for each different target can be replicated from a different multitarget particle.

An efficient solution has been proposed in the literature under a *posterior independence* assumption that unfortunately is almost never met in practical situations. This work provides another solution that does not rely on the posterior independence assumption and that is based on introducing an auxiliary Markov transition in the design. This approach can be seen as an extension of the auxiliary particle filter, and may be of independent interest, outside the application to crossover and to multitarget tracking. Optimization of the design parameters is also addressed.

Keywords: Bayesian filtering, particle filtering, mutation/selection/crossover, optimal design, convex optimization, stochastic approximation

1 Introduction

To fix ideas, consider a hidden Markov model, i.e. a hidden (not observed) Markov chain $\{X_k\}$ with model transition kernel

$$Q_k(x, dx') = \mathbb{P}[X_k \in dx' \mid X_{k-1} = x] ,$$

and an observation sequence $\{Y_k\}$ related to hidden states, e.g.

$$Y_k = h_k(X_k) + V_k \quad \text{with} \quad V_k \sim q_k^V(v) dv ,$$

which defines a likelihood function

$$g_k(x') = q_k^V(Y_k - h_k(x')) ,$$

up to a normalizing constant. The solution to the Bayesian or MMSE estimation relies on the Bayesian filter

$$\mu_k(dx') = \mathbb{P}[X_k \in dx' \mid Y_0 \cdots Y_k] ,$$

i.e. on the conditionnal distribution of the current state X_k given past observations $(Y_0, \dots, Y_k)$. The Bayesian filter satisfies a simple recurrent equation

$$\mu_{k-1} \xrightarrow{\text{prediction}} \eta_k = \mu_{k-1} Q_k \xrightarrow{\text{correction}} \mu_k = g_k \cdot \eta_k$$

where the predictor

$$\eta_k(dx') = \mu_{k-1} Q_k(dx') = \int_E \mu_{k-1}(dx) Q_k(x, dx') ,$$

is obtained by acting the transition kernel $Q_k(x, dx')$ on the filter $\mu_{k-1}(dx)$, and the filter

$$\mu_k(dx') = (g_k \cdot \eta_k)(dx') \propto g_k(x') \eta_k(dx') ,$$

is obtained by applying the Bayes rule with the predictor $\eta_k(dx')$ and the likelihood function $g_k(x')$. The idea behind particle filtering, or sequential Monte Carlo methods, is to look for an approximation of the Bayesian filter as a weighted empirical probability distribution

$$\mu_k \approx \mu_k^N = \sum_{i=1}^N w_k^i \delta_{\xi_k^i} \quad \text{with} \quad \sum_{i=1}^N w_k^i = 1 ,$$

associated with a population of N particles characterized by their *positions* $(\xi_k^1, \dots, \xi_k^N)$ in E , and their non-negative normalized *weights* $(w_k^1, \dots, w_k^N)$. Plugging the particle approximation

$$\mu_{k-1}^N = \sum_{i=1}^N w_{k-1}^i \delta_{\xi_{k-1}^i} ,$$

in the recurrent equation, one obtains the finite mixture

$$\mu_{k-1}^N Q_k(dx') = \sum_{i=1}^N w_{k-1}^i Q_k(\xi_{k-1}^i, dx') ,$$

as an approximation of the predictor $\eta_k = \mu_{k-1} Q_k$, and the probability distribution (up to a normalizing constant)

$$g_k(x') \sum_{i=1}^N w_{k-1}^i Q_k(\xi_{k-1}^i, dx') ,$$

as an approximation of the filter $\mu_k = g_k \cdot \eta_k$. To further approximate the 'target' probability distribution

$$\underbrace{g_k(x')}_{g(x')} \underbrace{\sum_{i=1}^N w_{k-1}^i Q_k(\xi_{k-1}^i, dx')}_{\eta(dx')} \quad \text{with} \quad \sum_{i=1}^N w_{k-1}^i = 1 ,$$

seen as a Boltzmann–Gibbs probability distribution, where $\eta(dx')$ is a mixture probability distribution, the simplest importance sampling approximation is obtained as

$$\mu_k^N = \sum_{i=1}^N w_k^i \delta_{\xi_k^i} ,$$

where independently for any $i = 1, \dots, N$ the i -th particle ξ_k^i is sampled from $\eta(dx')$ and is assigned a weight w_k^i using $g(x')$. This is just a possible approximation, and it is well known that most sequential Monte Carlo methods exhibit a similar pattern and can be interpreted in terms of implementing *selection* and *mutation* steps, using the language of evolutionary algorithms. However, most general evolutionary algorithms also include a *crossover* step that is not present in sequential Monte Carlo methods. It is the objective of this work to make it possible to include a crossover step in sequential Monte Carlo methods.

A prototypical situation where crossover is needed, or at least could be useful, is multitarget tracking. In multitarget tracking, many particle approximations are available to sample from the filtering probability distribution, with the effect that a new multitarget particle is obtained by replicating *globally* an existing multitarget particle, i.e. in the new multitarget particle, the particle for each different target is replicated from the same multitarget particle. However, it may happen that a multitarget particle provides a good proxy for some targets and a bad proxy for the other targets, i.e. it may happen that some targets in a multitarget particle are good proxies, but are not going to be selected just because the other targets in the same multitarget particle are bad proxies. In other words, a multitarget particle is likely to be selected only if jointly all the targets deserve to be selected. This is unfair, and a better design would be to produce shuffled multitarget particles such that the particle for each different target can be replicated from a different multitarget particle. An efficient solution has been proposed by Ubéda–Medina et al. [5] under a *posterior independence* assumption, see Yi et al. [6], that unfortunately is almost never met in practical situations. The objective of this work is to propose another solution that does not rely on the posterior independence assumption and that is based on introducing an auxiliary Markov transition. This approach can be seen as an extension of the auxiliary particle filter, see Pitt and Shephard [4], and may be of independent interest, outside the application to crossover and to multitarget tracking.

This paper contains several sections, with many repetitions between the first four sections. Section 2 reviews the auxiliary particle filter. The next two sections, Section 3 and Section 4, consider the effect of adding a Markov transition in the design, using an auxiliary transition matrix and an auxiliary transition kernel, respectively. Section 5 uses this additional degree of freedom so as to introduce crossover between particles in association with a given partition of the state vector. Each of these four sections is organized along the same lines

- interpret the target probability distribution as the marginal of another probability distribution defined on a suitable *augmented* state space,

- design a class of particle filters on the augmented state space, based on a suitable *importance decomposition* and on the *importance sampling* principle,
- check that the more general class of particle filters contains known (and less general) classes of particle filters as special cases, i.e. *generalization* holds,
- check that the particle approximation of the normalization constant is an unbiased estimator, provide an expression of the variance of the approximation, characterize the *optimum design parameters*, and evaluate the associated minimum variance.

The last two sections, Section 6 and Section 7, consider two special cases where optimization of the design parameters is made simple, if not explicit, and the existence of a *unique* minimizer can be proved.

2 Basics on the auxiliary particle filter

Consider the probability distribution

$$\mu(dx') \propto \underbrace{g_k(x')}_{g(x')} \underbrace{\sum_{i=1}^N w_{k-1}^i Q_k(\xi_{k-1}^i, dx')}_{\eta(dx')} \quad \text{with} \quad \sum_{i=1}^N w_{k-1}^i = 1 ,$$

defined on E , seen as a Boltzmann–Gibbs probability distribution, where $\eta(dx')$ is a mixture probability distribution, and its importance sampling approximation

$$\mu^N = \sum_{i=1}^N w_k^i \delta_{\xi_k^i} ,$$

where independently for any $i = 1, \dots, N$ the i -th particle ξ_k^i is sampled from $\eta(dx')$ and is assigned a weight w_k^i using $g(x')$. In practice, independently for any $i = 1, \dots, N$

- the index a_k^i is sampled from the probability vector $(w_{k-1}^1, \dots, w_{k-1}^N)$,

then, setting $a = a_k^i$

- the particle ξ_k^i is sampled from the probability distribution $Q_k(\xi_{k-1}^a, dx')$,
- and it receives the weight $w_k^i \propto g(\xi_k^i)$, i.e.

$$w_k^i \propto g_k(\xi_k^i) .$$

The idea behind auxiliary particle filtering is to see $\mu(dx')$ as the marginal probability distribution on E of the joint probability distribution

$$\mu_{\text{aux}}^a(dx') \propto g_k(x') w_{k-1}^a Q_k(\xi_{k-1}^a, dx') ,$$

defined on the augmented space $\{1, \dots, N\} \times E$. Indeed, summation with respect to the index $a \in \{1, \dots, N\}$ yields

$$g_k(x') \sum_{a=1}^N w_{k-1}^a Q_k(\xi_{k-1}^a, dx') = g(x') \eta(dx') , \quad (1)$$

an identity for unnormalized distributions, that carries over to normalized probability distributions. Introducing the auxiliary probability vector $(\lambda_{\text{aux}}^1, \dots, \lambda_{\text{aux}}^N)$, the components of which are known as the *auxiliary weights*, makes it possible to define the importance decomposition

$$\mu_{\text{aux}}^a(dx') \propto \underbrace{g_k(x') \frac{w_{k-1}^a}{\lambda_{\text{aux}}^a}}_{g_{\text{aux}}^a(x')} \underbrace{Q_k(\xi_{k-1}^a, dx')}_{\eta_{\text{aux}}^a(dx')} , \quad (2)$$

the Monte Carlo approximation for the predictor

$$\eta_{\text{aux}}^N = \frac{1}{N} \sum_{i=1}^N \delta_{(a_k^i, \xi_k^i)} \quad \text{hence} \quad \langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle = \frac{1}{N} \sum_{i=1}^N g_{\text{aux}}^{a_k^i}(\xi_k^i) ,$$

and the importance sampling approximation for the filter

$$\mu_{\text{aux}}^N = \sum_{i=1}^N w_k^i \delta_{(a_k^i, \xi_k^i)} \quad \text{and its marginal} \quad \mu^N = \sum_{i=1}^N w_k^i \delta_{\xi_k^i} ,$$

where independently for any $i = 1, \dots, N$ the i -th particle (a_k^i, ξ_k^i) is sampled from $\eta_{\text{aux}}^a(dx')$ and is assigned a weight w_k^i using $g_{\text{aux}}^a(x')$. In practice, independently for any $i = 1, \dots, N$

- the index a_k^i is sampled from the auxiliary probability vector $(\lambda_{\text{aux}}^1, \dots, \lambda_{\text{aux}}^N)$,

then, setting $a = a_k^i$

- the particle ξ_k^i is sampled from the probability distribution $Q_k(\xi_{k-1}^a, dx')$,
- and it receives the weight $w_k^i \propto g_{\text{aux}}^a(\xi_k^i)$, i.e.

$$w_k^i \propto g_k(\xi_k^i) \frac{w_{k-1}^a}{\lambda_{\text{aux}}^a} ,$$

see Algorithm 1.

This class of particle filters depends upon $N-1$ design parameters, that are the components of the auxiliary probability vector λ_{aux} (a probability vector of dimension N) subject to normalization constraint. This class of particle filters contains the class of ordinary particle filters as a special case. Indeed, if $\lambda_{\text{aux}}^a = w_{k-1}^a$ for any $a \in \{1, \dots, N\}$, then the resulting particle filter reduces to the ordinary (non auxiliary) particle filter.

Recall the particle approximation

$$\langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle = \frac{1}{N} \sum_{i=1}^N g_{\text{aux}}^{a_k^i}(\xi_k^i) .$$

Algorithm 1: Particle filter with auxiliary weights

input : $\lambda_{\text{aux}}, (\xi_{k-1}^1, \dots, \xi_{k-1}^N), (w_{k-1}^1, \dots, w_{k-1}^N), Y_k$
output: $(\xi_k^1, \dots, \xi_k^N), (w_k^1, \dots, w_k^N)$
for $i = 1 \dots N$ **do**
 Sample the auxiliary variable
 Sample a from the auxiliary probability vector $\lambda_{\text{aux}} = (\lambda_{\text{aux}}^1, \dots, \lambda_{\text{aux}}^N)$
 Propagate the state vector
 Sample $\xi_k^i \sim Q_k(\xi_{k-1}^a, dx')$
 Evaluate the unnormalized weight
 Set $w_k^i = g_k(\xi_k^i) \frac{w_{k-1}^a}{\lambda_{\text{aux}}^a}$
end
 Normalize the weights

for the normalizing constant $\langle \eta_{\text{aux}}, g_{\text{aux}} \rangle$, where the random variables (a_k^i, ξ_k^i) for $i = 1, \dots, N$ are i.i.d. with common probability distribution $\eta_{\text{aux}}^a(dx')$, hence

$$\mathbb{E}[\langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle] = \langle \eta_{\text{aux}}, g_{\text{aux}} \rangle ,$$

i.e. the approximation is unbiased, and

$$N \mathbb{E} |\langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle - \langle \eta_{\text{aux}}, g_{\text{aux}} \rangle|^2 = \text{var}(g_{\text{aux}}, \eta_{\text{aux}}) = \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle - \langle \eta_{\text{aux}}, g_{\text{aux}} \rangle^2 .$$

Remark 2.1 Using the identity (1) and the importance decomposition (2) yields

$$\sum_{a=1}^N \int_E g_{\text{aux}}^a(x') \eta_{\text{aux}}^a(dx') = g(x') \eta(dx') ,$$

an identity for unnormalized distributions, that implies identity for normalizing constants. Indeed, integration of both sides with respect to the variable $x' \in E$ shows that the normalizing constant $\langle \eta_{\text{aux}}, g_{\text{aux}} \rangle$ associated with the importance decomposition (2) coincides with the normalizing constant $\langle \eta, g \rangle$.

Remark 2.2 Actually, $\langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle$ provides also an unbiased approximation of the normalizing constant $\langle \eta, g \rangle$, and the variance of the approximation error satisfies

$$N \mathbb{E} |\langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle - \langle \eta, g \rangle|^2 = \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle - \langle \eta, g \rangle^2 ,$$

hence minimizing this expression w.r.t. the auxiliary weights, reduces to minimizing $\langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle$ w.r.t. the auxiliary weights. Clearly, the minimum value is smaller than the value obtained for any choice of the auxiliary weights, and in particular is smaller than the value obtained for the special choice corresponding to the ordinary (non auxiliary) particle filter, hence

$$\min_{\lambda_{\text{aux}}} \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle \leq \langle \eta, g^2 \rangle \quad \text{and} \quad \min_{\lambda_{\text{aux}}} \text{var}(g_{\text{aux}}, \eta_{\text{aux}}) \leq \text{var}(g, \eta) .$$

Lemma 2.3 *Let $x = (x_1, \dots, x_N)$ be a non-zero vector of nonnegative real numbers. Minimizing the expression*

$$\sum_{a=1}^N \frac{x_a^2}{p_a} ,$$

over all probability vectors $p = (p_1, \dots, p_N)$ has a unique solution p proportional to x , and the minimum value is

$$\min_p \sum_{a=1}^N \frac{x_a^2}{p_a} = \left[\sum_{a=1}^N x_a \right]^2 .$$

PROOF. The problem is reformulated as a constrained optimization problem

$$\min_{p_1, \dots, p_N} \sum_{a=1}^N \frac{x_a^2}{p_a} \quad \text{subject to} \quad \sum_{a=1}^N p_a = 1 ,$$

the Lagrange multiplier μ is introduced and the Lagrangian

$$\sum_{a=1}^N \frac{x_a^2}{p_a} + \mu \left(\sum_{a=1}^N p_a - 1 \right) ,$$

is considered. First order optimality condition yields

$$-\left(\frac{x_a}{p_a}\right)^2 + \mu = 0 \quad \text{hence} \quad p_a = c x_a ,$$

for any $a \in \{1, \dots, N\}$. The constant c must be such that the constraint is satisfied, i.e.

$$p_a^{\text{opt}} = \frac{x_a}{\sum_{b=1}^N x_b} ,$$

for any $a \in \{1, \dots, N\}$. Plugging this expression yields

$$\min_p \sum_{a=1}^N \frac{x_a^2}{p_a} = \sum_{a=1}^N \frac{x_a^2}{p_a^{\text{opt}}} = \left[\sum_{a=1}^N \frac{x_a^2}{x_a} \right] \left[\sum_{b=1}^N x_b \right] = \left[\sum_{a=1}^N x_a \right]^2 . \quad \square$$

Let

$$u_a = \left\{ \int_E |g_k(x')|^2 Q_k(\xi_{k-1}^a, dx') \right\}^{1/2} . \quad (3)$$

The optimal design, that minimizes the variance of $\langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle$ seen as an unbiased approximation of the normalizing constant $\langle \eta, g \rangle$, is defined as follows.

Proposition 2.4 *For any $a \in \{1 \dots N\}$, the optimal choice for the auxiliary weight is*

$$\lambda_{\text{opt}}^a \propto w_{k-1}^a u_a ,$$

and the minimum value is

$$\min_{\lambda_{\text{aux}}} \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle = \left[\sum_{a=1}^N w_{k-1}^a u_a \right]^2 .$$

PROOF. The variance of the approximation error is controlled by

$$\begin{aligned}
\langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle &= \sum_{a=1}^N \int_E |g_{\text{aux}}^a(x')|^2 \eta_{\text{aux}}^a(dx') \\
&= \sum_{a=1}^N \int_E \left| \frac{g_k(x') w_{k-1}^a}{\lambda_{\text{aux}}^a} \right|^2 \lambda_{\text{aux}}^a Q_k(\xi_{k-1}^a, dx') \\
&= \sum_{a=1}^N \frac{|w_{k-1}^a|^2}{\lambda_{\text{aux}}^a} \int_E |g_k(x')|^2 Q_k(\xi_{k-1}^a, dx') \\
&= \sum_{a=1}^N \frac{|w_{k-1}^a|^2}{\lambda_{\text{aux}}^a} |u_a|^2,
\end{aligned}$$

and the result follows from Lemma 2.3. $\square$

Remark 2.5 It follows from the Cauchy–Schwartz inequality that

$$\left[\sum_{a=1}^N w_{k-1}^a u_a \right]^2 \leq \sum_{a=1}^N w_{k-1}^a |u_a|^2,$$

and note that

$$\sum_{a=1}^N w_{k-1}^a |u_a|^2 = \sum_{a=1}^N w_{k-1}^a \int_E |g_k(x')|^2 Q_k(\xi_{k-1}^a, dx') = \langle \eta, g^2 \rangle.$$

In other words, the optimal design for the auxiliary particle filter improves the performance (reduces the variance) over the classical particle filter.

Under the optimal design, independently for any $i \in \{1, \dots, N\}$

- the index a_k^i is sampled from the optimal probability vector $(\lambda_{\text{opt}}^1, \dots, \lambda_{\text{opt}}^N)$,

then, setting $a = a_k^i$

- the particle ξ_k^i is sampled from the probability distribution $Q_k(\xi_{k-1}^a, dx')$,
- and it receives the weight $w_k^i \propto g_{\text{opt}}^a(\xi_k^i)$, i.e.

$$w_k^i \propto g_k(\xi_k^i) \frac{w_{k-1}^a}{\lambda_{\text{opt}}^a} \propto \frac{1}{u_a} g_k(\xi_k^i),$$

see Algorithm 2.

To implement this optimal design, the challenge is to compute

$$u_a = \left\{ \int_E |g_k(x')|^2 Q_k(\xi_{k-1}^a, dx') \right\}^{1/2},$$

which was defined in (3), so as to set the optimal probability vector λ_{opt}^a , for any $a \in \{1, \dots, N\}$.

Algorithm 2: Particle filter with optimal auxiliary weights

input : $(\xi_{k-1}^1, \dots, \xi_{k-1}^N), (w_{k-1}^1, \dots, w_{k-1}^N), Y_k$
output: $(\xi_k^1, \dots, \xi_k^N), (w_k^1, \dots, w_k^N)$
for $a = 1 \dots N$ **do**
 Optimize the design parameter
 Compute u_a
 Evaluate the unnormalized optimal auxiliary weight
 Set $\lambda_{\text{opt}}^a = w_{k-1}^a u_a$
end
 Normalize the optimal auxiliary weights
for $i = 1 \dots N$ **do**
 Sample the auxiliary variable
 Sample a from the optimal probability vector $\lambda_{\text{opt}} = (\lambda_{\text{opt}}^1, \dots, \lambda_{\text{opt}}^N)$
 Propagate the state vector
 Sample $\xi_k^i \sim Q_k(\xi_{k-1}^a, dx')$
 Evaluate the unnormalized weight
 Set $w_k^i = \frac{1}{u_a} g_k(\xi_k^i)$
end
 Normalize the weights

3 Particle filter with an auxiliary transition matrix

Introducing the auxiliary $N \times N$ transition probability matrix $\pi_{\text{aux}} = (\pi_{\text{aux}}^{i,j})$ from $\{1, \dots, N\}$ to $\{1, \dots, N\}$, equivalently seen as a collection indexed by $i \in \{1, \dots, N\}$ of probability vectors on $\{1, \dots, N\}$, the idea is to see $\mu(dx')$ as the marginal probability distribution on E of the joint probability distribution

$$\mu_{\text{aux}}^{a,c}(dx') \propto g_k(x') w_{k-1}^a \pi_{\text{aux}}^{a,c} Q_k(\xi_{k-1}^a, dx') ,$$

defined on the augmented space $\{1, \dots, N\} \times \{1, \dots, N\} \times E$. Indeed, summation with respect to the indices $a \in \{1, \dots, N\}$ and $c \in \{1, \dots, N\}$ yields

$$g_k(x') \sum_{a=1}^N w_{k-1}^a \left[\sum_{c=1}^N \pi_{\text{aux}}^{a,c} \right] Q_k(\xi_{k-1}^a, dx') = g(x') \eta(dx') , \quad (4)$$

an identity for unnormalized distributions, that carries over to normalized probability distributions. Assuming that the model transition kernels have a density, i.e. assuming that

$$Q_k(x, dx') = q_k(x, x') dx' ,$$

makes it possible to define the importance decomposition

$$\mu_{\text{aux}}^{a,c}(dx') \propto g_k(x') \underbrace{\frac{q_k(\xi_{k-1}^a, x')}{q_k(\xi_{k-1}^c, x')} \frac{w_{k-1}^a}{\lambda_{\text{aux}}^a}}_{g_{\text{aux}}^{a,c}(x')} \underbrace{\lambda_{\text{aux}}^a \pi_{\text{aux}}^{a,c} Q_k(\xi_{k-1}^c, dx')}_{\eta_{\text{aux}}^{a,c}(dx')} , \quad (5)$$

the Monte Carlo approximation for the predictor

$$\eta_{\text{aux}}^N = \frac{1}{N} \sum_{i=1}^N \delta_{(a_k^i, c_k^i, \xi_k^i)} \quad \text{hence} \quad \langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle = \frac{1}{N} \sum_{i=1}^N g_{\text{aux}}^{a_k^i, c_k^i}(\xi_k^i),$$

and the importance sampling approximation for the filter

$$\mu_{\text{aux}}^N = \sum_{i=1}^N w_k^i \delta_{(a_k^i, c_k^i, \xi_k^i)} \quad \text{and its marginal} \quad \mu^N = \sum_{i=1}^N w_k^i \delta_{\xi_k^i},$$

where independently for any $i = 1, \dots, N$ the i -th particle (a_k^i, c_k^i, ξ_k^i) is sampled from $\eta_{\text{aux}}^{a,c}(dx')$ and is assigned a weight w_k^i using $g_{\text{aux}}^{a,c}(x')$. In practice, independently for any $i = 1, \dots, N$

- the index a_k^i is sampled from the auxiliary probability vector $(\lambda_{\text{aux}}^1, \dots, \lambda_{\text{aux}}^N)$,

then, setting $a = a_k^i$

- the index c_k^i is sampled from the probability vector $(\pi_{\text{aux}}^{a,1}, \dots, \pi_{\text{aux}}^{a,N})$, the a -th row of the auxiliary transition probability matrix π_{aux} ,

then, setting $c = c_k^i$

- the particle ξ_k^i is sampled from the probability distribution $Q_k(\xi_{k-1}^c, dx')$,
- and it receives the weight $w_k^i \propto g_{\text{aux}}^{a,c}(\xi_k^i)$, i.e.

$$w_k^i \propto g_k(\xi_k^i) \frac{q_k(\xi_{k-1}^a, \xi_k^i)}{q_k(\xi_{k-1}^c, \xi_k^i)} \frac{w_{k-1}^a}{\lambda_{\text{aux}}^a},$$

see Algorithm 3.

This class of particle filters depends upon $(N-1) + N(N-1) = N^2 - 1$ design parameters, that are the components of the auxiliary probability vector λ_{aux} (a probability vector of dimension N), and the components of the N rows of the auxiliary transition matrix π_{aux} (N probability vectors of dimension N each), subject to normalization constraints. This class of particle filters contains the class of auxiliary particle filters — and a fortiori it contains the class of ordinary (non auxiliary) particle filters — as a special case. Indeed

- if for any $a \in \{1, \dots, N\}$ the a -th row of the auxiliary transition matrix π_{aux} satisfies $\pi_{\text{aux}}^{a,c} = 1$ if $c = a$ and $\pi_{\text{aux}}^{a,c} = 0$ otherwise, then the resulting particle filter belongs to the class of auxiliary particle filters described in Section 2,
- moreover, if $\lambda_{\text{aux}}^a = w_{k-1}^a$ for any $a \in \{1, \dots, N\}$, then it further reduces to the ordinary (non auxiliary) particle filter algorithm.

Algorithm 3: Particle filter with auxiliary transition matrix

input : $\lambda_{\text{aux}}, \pi_{\text{aux}}, (\xi_{k-1}^1, \dots, \xi_{k-1}^N), (w_{k-1}^1, \dots, w_{k-1}^N), Y_k$
output: $(\xi_k^1, \dots, \xi_k^N), (w_k^1, \dots, w_k^N)$
for $i = 1 \dots N$ **do**
 Sample the auxiliary variable
 Sample a from the auxiliary probability vector $\lambda_{\text{aux}} = (\lambda_{\text{aux}}^1, \dots, \lambda_{\text{aux}}^N)$
 Sample the shuffling index
 Sample c from the a -th row of the auxiliary transition matrix π_{aux} , i.e. from the
 probability vector $(\pi_{\text{aux}}^{a,1}, \dots, \pi_{\text{aux}}^{a,N})$
 Propagate the state vector
 Sample $\xi_k^i \sim Q_k(\xi_{k-1}^c, dx')$
 Evaluate the unnormalized weight
 Set $w_k^i = g_k(\xi_k^i) \frac{w_{k-1}^a}{\lambda_{\text{aux}}^a} \frac{q_k(\xi_{k-1}^a, \xi_k^i)}{q_k(\xi_{k-1}^c, \xi_k^i)}$
end
 Normalize the weights

Recall the particle approximation

$$\langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle = \frac{1}{N} \sum_{i=1}^N g_{\text{aux}}^{a_k^i, c_k^i}(\xi_k^i),$$

for the normalizing constant $\langle \eta_{\text{aux}}, g_{\text{aux}} \rangle$, where the random variables (a_k^i, c_k^i, ξ_k^i) for $i = 1, \dots, N$ are i.i.d. with common probability distribution $\eta_{\text{aux}}^{a,c}(dx')$, hence

$$\mathbb{E}[\langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle] = \langle \eta_{\text{aux}}, g_{\text{aux}} \rangle,$$

i.e. the approximation is unbiased, and

$$N \mathbb{E} |\langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle - \langle \eta_{\text{aux}}, g_{\text{aux}} \rangle|^2 = \text{var}(g_{\text{aux}}, \eta_{\text{aux}}) = \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle - \langle \eta_{\text{aux}}, g_{\text{aux}} \rangle^2.$$

Remark 3.1 Using the identity (4) and the importance decomposition (5) yields

$$\sum_{a=1}^N \sum_{c=1}^N g_{\text{aux}}^{a,c}(x') \eta_{\text{aux}}^{a,c}(dx') = g(x') \eta(dx'),$$

an identity for unnormalized distributions, that implies identity for normalizing constants. Indeed, integration of both sides with respect to the variable $x' \in E$ shows that the normalizing constant $\langle \eta_{\text{aux}}, g_{\text{aux}} \rangle$ associated with the importance decomposition (5) coincides with the normalizing constant $\langle \eta, g \rangle$.

Remark 3.2 Actually, $\langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle$ provides also an unbiased approximation of the normalizing constant $\langle \eta, g \rangle$, and the variance of the approximation error satisfies

$$N \mathbb{E} |\langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle - \langle \eta, g \rangle|^2 = \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle - \langle \eta, g \rangle^2.$$

hence minimizing this expression w.r.t. the auxiliary weights and the auxiliary transition matrix, reduces to minimizing $\langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle$ w.r.t. the auxiliary weights and the auxiliary transition matrix. Clearly, the minimum value is smaller than the value obtained for any choice of the auxiliary weights and the auxiliary transition matrix, and in particular is smaller than the value obtained for the special choice corresponding to the auxiliary particle filter or the ordinary (non auxiliary) particle filter, hence

$$\min_{(\lambda_{\text{aux}}, \pi_{\text{aux}})} \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle \leq \langle \eta, g^2 \rangle \quad \text{and} \quad \min_{(\lambda_{\text{aux}}, \pi_{\text{aux}})} \text{var}(g_{\text{aux}}, \eta_{\text{aux}}) \leq \text{var}(g, \eta) .$$

Let

$$u_a^c = \left\{ \int_E |g_k(x')| \frac{q_k(\xi_{k-1}^a, x')}{q_k(\xi_{k-1}^c, x')} |^2 Q_k(\xi_{k-1}^c, dx') \right\}^{1/2} , \quad (6)$$

and note that in the special case where $c = a$, this expression reduces to

$$u_a^a = \left\{ \int_E |g_k(x')|^2 Q_k(\xi_{k-1}^a, dx') \right\}^{1/2} = u_a ,$$

which was defined in (3). The optimal design, that minimizes the variance of $\langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle$ seen as an unbiased approximation of the normalizing constant $\langle \eta, g \rangle$, is defined as follows.

Proposition 3.3 *For any $a \in \{1 \cdots N\}$, let $u_a^\bullet$ denote the minimum value of u_a^c when $c \in \{1 \cdots N\}$ and assume that there exists a unique minimizer, i.e. a unique index $c_a^\bullet$ for which the minimum value is achieved. Then for any $a \in \{1 \cdots N\}$ the optimal choice for the a -th row of the auxiliary transition matrix is such that*

$$\pi_{\text{opt}}^{a,c} = 1 \text{ if } c = c_a^\bullet \text{ and } \pi_{\text{opt}}^{a,c} = 0 \text{ otherwise,}$$

the optimal choice for the auxiliary weight is

$$\lambda_{\text{opt}}^a \propto w_{k-1}^a u_a^\bullet ,$$

and the minimum value is

$$\min_{(\lambda_{\text{aux}}, \pi_{\text{aux}})} \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle = \left[\sum_{a=1}^N w_{k-1}^a u_a^\bullet \right]^2 .$$

PROOF. The variance of the approximation error is controlled by

$$\begin{aligned} \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle &= \sum_{a=1}^N \sum_{c=1}^N \int_E |g_{\text{aux}}^{a,c}(x')|^2 \eta_{\text{aux}}^{a,c}(dx') \\ &= \sum_{a=1}^N \sum_{c=1}^N \int_E |g_k(x')| \frac{q_k(\xi_{k-1}^a, x')}{q_k(\xi_{k-1}^c, x')} \frac{w_{k-1}^a}{\lambda_{\text{aux}}^a} |^2 \lambda_{\text{aux}}^a \pi_{\text{aux}}^{a,c} Q_k(\xi_{k-1}^c, dx') \\ &= \sum_{a=1}^N \frac{|w_{k-1}^a|^2}{\lambda_{\text{aux}}^a} \sum_{c=1}^N \pi_{\text{aux}}^{a,c} \int_E |g_k(x')| \frac{q_k(\xi_{k-1}^a, x')}{q_k(\xi_{k-1}^c, x')} |^2 Q_k(\xi_{k-1}^c, dx') \\ &= \sum_{a=1}^N \frac{|w_{k-1}^a|^2}{\lambda_{\text{aux}}^a} \sum_{c=1}^N \pi_{\text{aux}}^{a,c} |u_a^c|^2 . \end{aligned}$$

Under the assumptions, it holds

$$\sum_{c=1}^N \pi_{\text{aux}}^{a,c} |u_a^c|^2 \geq |u_a^\bullet|^2 ,$$

and the lower bound is achieved if the a -th row of the auxiliary transition matrix π_{aux} charges the minimizer only, i.e. if it satisfies $\pi_{\text{aux}}^{a,c} = 1$ if $c = c_a^\bullet$ and $\pi_{\text{aux}}^{a,c} = 0$ otherwise. Plugging this expression yields

$$\begin{aligned} \min_{\pi_{\text{aux}}} \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle &= \min_{\pi_{\text{aux}}} \sum_{a=1}^N \frac{|w_{k-1}^a|^2}{\lambda_{\text{aux}}^a} \sum_{c=1}^N \pi_{\text{aux}}^{a,c} |u_a^c|^2 \\ &= \sum_{a=1}^N \frac{|w_{k-1}^a|^2}{\lambda_{\text{aux}}^a} \sum_{c=1}^N \pi_{\text{opt}}^{a,c} |u_a^c|^2 \\ &= \sum_{a=1}^N \frac{|w_{k-1}^a|^2}{\lambda_{\text{aux}}^a} |u_a^\bullet|^2 , \end{aligned}$$

and it follows from Lemma 2.3 that the minimum w.r.t. the auxiliary probability vector is achieved for

$$\lambda_{\text{opt}}^a \propto w_{k-1}^a u_a^\bullet ,$$

and

$$\begin{aligned} \min_{(\lambda_{\text{aux}}, \pi_{\text{aux}})} \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle &= \min_{\lambda_{\text{aux}}} \min_{\pi_{\text{aux}}} \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle \\ &= \min_{\lambda_{\text{aux}}} \sum_{a=1}^N \frac{|w_{k-1}^a u_a^\bullet|^2}{\lambda_{\text{aux}}^a} \\ &= \left[\sum_{a=1}^N w_{k-1}^a u_a^\bullet \right]^2 . \quad \square \end{aligned}$$

Remark 3.4 Recall that $u_a^\bullet$ denotes the minimum value of u_a^c when $c \in \{1, \dots, N\}$ and in particular $u_a^\bullet \leq u_a^a = u_a$, hence

$$\left[\sum_{a=1}^N w_{k-1}^a u_a^\bullet \right]^2 \leq \left[\sum_{a=1}^N w_{k-1}^a u_a \right]^2 ,$$

for any $a \in \{1, \dots, N\}$. In other words, the optimal design for the particle filter with an auxiliary transition matrix improves the performance (reduces the variance) over the optimal design for the auxiliary particle filter, and a fortiori it improves the performance over the ordinary (non auxiliary) particle filter.

Under the optimal design, independently for any $i \in \{1, \dots, N\}$

- the index a_k^i is sampled from the optimal probability vector $(\lambda_{\text{opt}}^1, \dots, \lambda_{\text{opt}}^N)$,

then, setting $a = a_k^i$ and $c = c_a^\bullet$

- the particle ξ_k^i is sampled from the probability distribution $Q_k(\xi_{k-1}^c, dx')$,
- and it receives the weight $w_k^i \propto g_{\text{opt}}^{a,c}(\xi_k^i)$, i.e.

$$w_k^i \propto g_k(\xi_k^i) \frac{q_k(\xi_{k-1}^a, \xi_k^i)}{q_k(\xi_{k-1}^c, \xi_k^i)} \frac{w_{k-1}^a}{\lambda_{\text{opt}}^a} \propto \frac{1}{u_a^\bullet} g_k(\xi_k^i) \frac{q_k(\xi_{k-1}^a, \xi_k^i)}{q_k(\xi_{k-1}^c, \xi_k^i)} .$$

see Algorithm 4.

Algorithm 4: Particle filter with optimal auxiliary transition matrix

input : $(\xi_{k-1}^1, \dots, \xi_{k-1}^N), (w_{k-1}^1, \dots, w_{k-1}^N), Y_k$
output: $(\xi_k^1, \dots, \xi_k^N), (w_k^1, \dots, w_k^N)$
for $a = 1 \dots N$ **do**
 Optimize the design parameters
 Compute the minimum value $u_a^\bullet$ and the minimizer $c_a^\bullet$
 Evaluate the unnormalized optimal auxiliary weight
 Set $\lambda_{\text{opt}}^a = w_{k-1}^a u_a^\bullet$
end
Normalize the optimal auxiliary weights
for $i = 1 \dots N$ **do**
 Sample the auxiliary variable
 Sample a from the optimal probability vector $\lambda_{\text{opt}} = (\lambda_{\text{opt}}^1, \dots, \lambda_{\text{opt}}^N)$
 Propagate the state vector
 Set $c = c_a^\bullet$ and sample $\xi_k^i \sim Q_k(\xi_{k-1}^c, dx')$
 Evaluate the unnormalized weight
 Set $w_k^i = \frac{1}{u_a^\bullet} g_k(\xi_k^i) \frac{q_k(\xi_{k-1}^a, \xi_k^i)}{q_k(\xi_{k-1}^c, \xi_k^i)}$
end
Normalize the weights

To implement this optimal design, the challenge is to compute

$$u_a^c = \left\{ \int_E |g_k(x')| \frac{q_k(\xi_{k-1}^a, x')}{q_k(\xi_{k-1}^c, x')} |^2 Q_k(\xi_{k-1}^c, dx') \right\}^{1/2} ,$$

which was defined in (6), for any index $a \in \{1, \dots, N\}$ and any index $c \in \{1, \dots, N\}$, so as to compute the minimum value $u_a^\bullet$, find the minimizer $c_a^\bullet$ and set the optimal probability vector λ_{opt}^a , for any $a \in \{1, \dots, N\}$. Introduce the mapping

$$u_a(z) = \left\{ \int_E |g_k(x')| \frac{q_k(\xi_{k-1}^a, x')}{q_k(z, x')} |^2 Q_k(z, dx') \right\}^{1/2} , \quad (7)$$

and note that in the special case where $z = \xi_{k-1}^c$ this expression reduces to

$$u_a(\xi_{k-1}^c) = \left\{ \int_E |g_k(x')| \frac{q_k(\xi_{k-1}^a, x')}{q_k(\xi_{k-1}^c, x')} |^2 Q_k(\xi_{k-1}^c, dx') \right\}^{1/2} = u_a^c .$$

Remark 3.5 The problem of minimizing $u_a^c = u_a(\xi_{k-1}^c)$ with respect to the index $c \in \{1, \dots, N\}$ can only be solved by exhaustive search within a finite (but large) set. If this combinatorial minimization problem would be replaced by the wider problem of minimizing $u_a(z)$ with respect to the variable $z \in E$, as if there would be a continuum of possible choices and not a finite (but large) number N of possible choices, then it would be possible to use numerical optimization procedures instead of exhaustive search.

This motivates the introduction in Section 4 of another (and larger) class of particle filters, with an auxiliary transition kernel.

4 Particle filter with an auxiliary transition kernel

Introducing the auxiliary transition probability kernel $\kappa_{\text{aux}} = (\kappa_{\text{aux}}^i(dz))$ from $\{1, \dots, N\}$ to E , equivalently seen as a collection indexed by $i \in \{1, \dots, N\}$ of probability distributions on E , the idea is to see $\mu(dx')$ as the marginal probability distribution on E of the joint probability distribution

$$\mu_{\text{aux}}^a(dz, dx') \propto g_k(x') w_{k-1}^a \kappa_{\text{aux}}^a(dz) Q_k(\xi_{k-1}^a, dx'),$$

defined on the augmented space $\{1, \dots, N\} \times E \times E$. Indeed, summation with respect to the index $a \in \{1, \dots, N\}$ and integration with respect to the variable $z \in E$ yields

$$g_k(x') \sum_{a=1}^N w_{k-1}^a \left[\int_E \kappa_{\text{aux}}^a(dz) \right] Q_k(\xi_{k-1}^a, dx') = g(x') \eta(dx'), \quad (8)$$

an identity for unnormalized distributions, that carries over to normalized probability distributions. Assuming that the model transition kernels have a density, i.e. assuming that

$$Q_k(x, dx') = q_k(x, x') dx',$$

makes it possible to define the importance decomposition

$$\mu_{\text{aux}}^a(dz, dx') \propto g_k(x') \underbrace{\frac{q_k(\xi_{k-1}^a, x')}{q_k(z, x')} \frac{w_{k-1}^a}{\lambda_{\text{aux}}^a}}_{g_{\text{aux}}^a(z, x')} \underbrace{\lambda_{\text{aux}}^a \kappa_{\text{aux}}^a(dz) Q_k(z, dx')}_{\eta_{\text{aux}}^a(dz, dx')}, \quad (9)$$

the Monte Carlo approximation for the predictor

$$\eta_{\text{aux}}^N = \frac{1}{N} \sum_{i=1}^N \delta_{(a_k^i, \zeta_k^i, \xi_k^i)} \quad \text{hence} \quad \langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle = \frac{1}{N} \sum_{i=1}^N g_{\text{aux}}^i(\zeta_k^i, \xi_k^i),$$

and the importance sampling approximation for the filter

$$\mu_{\text{aux}}^N = \sum_{i=1}^N w_k^i \delta_{(a_k^i, \zeta_k^i, \xi_k^i)} \quad \text{and its marginal} \quad \mu^N = \sum_{i=1}^N w_k^i \delta_{\xi_k^i},$$

where independently for any $i = 1, \dots, N$ the i -th particle $(a_k^i, \zeta_k^i, \xi_k^i)$ is sampled from $\eta_{\text{aux}}^a(dz, dx')$ and is assigned a weight w_k^i using $g_{\text{aux}}^a(z, x')$. In practice, independently for any $i = 1, \dots, N$

- the index a_k^i is sampled from the auxiliary probability vector $(\lambda_{\text{aux}}^1, \dots, \lambda_{\text{aux}}^N)$,

then, setting $a = a_k^i$

- the particle ζ_k^i is sampled from the probability distribution $\kappa_{\text{aux}}^a(dz)$, the a -th row of the auxiliary transition probability kernel κ_{aux} ,

then, setting $z = \zeta_k^i$

- the particle ξ_k^i is sampled from the probability distribution $Q_k(z, dx')$,
- and it receives the weight $w_k^i \propto g_{\text{aux}}^a(z, \xi_k^i)$, i.e.

$$w_k^i \propto g_k(\xi_k^i) \frac{q_k(\xi_{k-1}^a, \xi_k^i)}{q_k(z, \xi_k^i)} \frac{w_{k-1}^a}{\lambda_{\text{aux}}^a},$$

see Algorithm 5.

Algorithm 5: Particle filter with auxiliary transition kernel

input : $\lambda_{\text{aux}}, \kappa_{\text{aux}}, (\xi_{k-1}^1, \dots, \xi_{k-1}^N), (w_{k-1}^1, \dots, w_{k-1}^N), Y_k$
output: $(\xi_k^1, \dots, \xi_k^N), (w_k^1, \dots, w_k^N)$
for $i = 1 \dots N$ **do**
 Sample the auxiliary variable
 Sample a from the auxiliary probability vector $\lambda_{\text{aux}} = (\lambda_{\text{aux}}^1, \dots, \lambda_{\text{aux}}^N)$
 Sample the shuffled state vector
 Sample z from the a -th 'row' of the auxiliary transition kernel κ_{aux} , i.e. from the probability distribution $\kappa_{\text{aux}}^a(dz)$
 Propagate the state vector
 Sample $\xi_k^i \sim Q_k(z, dx')$
 Evaluate the unnormalized weight
 Set $w_k^i = g_k(\xi_k^i) \frac{w_{k-1}^a}{\lambda_{\text{aux}}^a} \frac{q_k(\xi_{k-1}^a, \xi_k^i)}{q_k(z, \xi_k^i)}$
end
 Normalize the weights

This class of particle filters depends upon several design parameters, that are the components of the auxiliary probability vector λ_{aux} (a probability vector of dimension N) subject to normalization constraint, and the N 'rows' of the auxiliary transition kernel κ_{aux} (N probability distributions defined on E each). This class of particle filters contains the class of particle filters with an auxiliary transition matrix — and a fortiori it contains the class of auxiliary particle filters and the class of ordinary (non auxiliary) particle filters — as a special case. Indeed

- if for any $a \in \{1, \dots, N\}$ the a -th 'row' of the auxiliary transition kernel κ_{aux} charges the finite set of available particles only, i.e. if

$$\kappa_{\text{aux}}^a(dz) = \sum_{c=1}^N \pi_{\text{aux}}^{a,c} \delta_{\xi_{k-1}^c}(dz),$$

as a finite mixture, where the vector of mixture weights is given as the a -th row of some transition matrix π_{aux} of dimension $N \times N$, then the resulting particle filter belongs to the class of particle filters with an auxiliary transition matrix described in Section 3,

- moreover, if for any $a \in \{1, \dots, N\}$ the a -th row of the auxiliary transition matrix π_{aux} satisfies $\pi_{\text{aux}}^{a,c} = 1$ if $c = a$ and $\pi_{\text{aux}}^{a,c} = 0$ otherwise, then the a -th 'row' of the auxiliary transition kernel κ_{aux} charges the particle ξ_{k-1}^a only, i.e.

$$\kappa_{\text{aux}}^a(dz) = \delta_{\xi_{k-1}^a}^a(dz) ,$$

and the resulting particle filter belongs to the class of auxiliary particle filters described in Section 2,

- moreover, if $\lambda_{\text{aux}}^a = w_{k-1}^a$ for any $a \in \{1, \dots, N\}$, then it further reduces to the ordinary (non auxiliary) particle filter.

Recall the particle approximation

$$\langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle = \frac{1}{N} \sum_{i=1}^N g_{\text{aux}}^{a_k^i}(\zeta_k^i, \xi_k^i) .$$

for the normalizing constant $\langle \eta_{\text{aux}}, g_{\text{aux}} \rangle$, where the random variables $(a_k^i, \zeta_k^i, \xi_k^i)$ for $i = 1, \dots, N$ are i.i.d. with common probability distribution $\eta_{\text{aux}}^a(dz, dx')$, hence

$$\mathbb{E}[\langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle] = \langle \eta_{\text{aux}}, g_{\text{aux}} \rangle ,$$

i.e. the approximation is unbiased, and

$$N \mathbb{E} |\langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle - \langle \eta_{\text{aux}}, g_{\text{aux}} \rangle|^2 = \text{var}(g_{\text{aux}}, \eta_{\text{aux}}) = \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle - \langle \eta_{\text{aux}}, g_{\text{aux}} \rangle^2 .$$

Remark 4.1 Using the identity (8) and the importance decomposition (9) yields

$$\sum_{a=1}^N \int_E g_{\text{aux}}^a(z, x') \eta_{\text{aux}}^a(dz, dx') = g(x') \eta(dx') ,$$

an identity for unnormalized distributions, that implies identity for normalizing constants. Indeed, integration of both sides with respect to the variable $x' \in E$ shows that the normalizing constant $\langle \eta_{\text{aux}}, g_{\text{aux}} \rangle$ associated with the importance decomposition (9) coincides with the normalizing constant $\langle \eta, g \rangle$.

Remark 4.2 Actually, $\langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle$ provides also an unbiased approximation of the normalizing constant $\langle \eta, g \rangle$, and the variance of the approximation error satisfies

$$N \mathbb{E} |\langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle - \langle \eta, g \rangle|^2 = \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle - \langle \eta, g \rangle^2 ,$$

therefore minimizing this expression w.r.t. the auxiliary weights and the auxiliary transition kernel, reduces to minimizing $\langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle$ w.r.t. the auxiliary weights and the auxiliary transition kernel. Clearly, the minimum value is smaller than the value obtained for any choice of the

auxiliary weights and the auxiliary transition kernel, and in particular is smaller than the value obtained for the special choice corresponding to the particle filter with an auxiliary transition matrix, the auxiliary particle filter or the ordinary (non auxiliary) particle filter, hence

$$\min_{(\lambda_{\text{aux}}, \kappa_{\text{aux}})} \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle \leq \langle \eta, g^2 \rangle \quad \text{and} \quad \min_{(\lambda_{\text{aux}}, \kappa_{\text{aux}})} \text{var}(g_{\text{aux}}, \eta_{\text{aux}}) \leq \text{var}(g, \eta) .$$

Let

$$u_a(z) = \left\{ \int_E |g_k(x')| \frac{q_k(\xi_{k-1}^a, x')}{q_k(z, x')}^2 Q_k(z, dx') \right\}^{1/2} ,$$

which was defined in (7). The optimal design, that minimizes the variance of $\langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle$ seen as an unbiased approximation of the normalizing constant $\langle \eta, g \rangle$, is defined as follows.

Proposition 4.3 *For any $a \in \{1 \cdots N\}$, let $u_a^{\bullet\bullet}$ denote the minimum value of $u_a(z)$ when $z \in E$ and assume that there exists a unique minimizer, i.e. a unique vector $z_a^{\bullet\bullet}$ for which the minimum value is achieved. Then for any $a \in \{1 \cdots N\}$ the optimal choice for the a -th 'row' of the auxiliary transition kernel is*

$$\kappa_{\text{opt}}^a(dz) = \delta_{z_a^{\bullet\bullet}}(dz) ,$$

the optimal choice for the auxiliary weight is

$$\lambda_{\text{opt}}^a \propto w_{k-1}^a u_a^{\bullet\bullet} ,$$

and the minimum value is

$$\min_{(\lambda_{\text{aux}}, \kappa_{\text{aux}})} \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle = \left[\sum_{a=1}^N w_{k-1}^a u_a^{\bullet\bullet} \right]^2 .$$

PROOF. The variance of the approximation error is controlled by

$$\begin{aligned} \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle &= \sum_{a=1}^N \int_E \int_E |g_{\text{aux}}^a(z, x')|^2 \eta_{\text{aux}}^a(dz, dx') \\ &= \sum_{a=1}^N \int_E \int_E |g_k(x')| \frac{q_k(\xi_{k-1}^a, x')}{q_k(z, x')} \frac{w_{k-1}^a}{\lambda_{\text{aux}}^a}^2 \lambda_{\text{aux}}^a \kappa_{\text{aux}}^a(dz) Q_k(z, dx') \\ &= \sum_{a=1}^N \frac{|w_{k-1}^a|^2}{\lambda_{\text{aux}}^a} \int_E \left[\int_E |g_k(x')| \frac{q_k(\xi_{k-1}^a, x')}{q_k(z, x')}^2 Q_k(z, dx') \right] \kappa_{\text{aux}}^a(dz) \\ &= \sum_{a=1}^N \frac{|w_{k-1}^a|^2}{\lambda_{\text{aux}}^a} \int_E |u_a(z)|^2 \kappa_{\text{aux}}^a(dz) . \end{aligned}$$

Under the assumptions, it holds

$$\int_E |u_a(z)|^2 \kappa_{\text{aux}}^a(dz) \geq |u_a^{\bullet\bullet}|^2 ,$$

and the lower bound is achieved if the a -th 'row' of the transition probability kernel κ_{aux} charges the minimizer only, i.e. if it satisfies $\kappa_{\text{aux}}^a(dz) = \delta_{z_a^{\bullet\bullet}}(dz)$. Plugging this expression yields

$$\begin{aligned} \min_{\kappa_{\text{aux}}} \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle &= \min_{\kappa_{\text{aux}}} \sum_{a=1}^N \frac{|w_{k-1}^a|^2}{\lambda_{\text{aux}}^a} \int_E |u_a(z)|^2 \kappa_{\text{aux}}^a(dz) \\ &= \sum_{a=1}^N \frac{|w_{k-1}^a|^2}{\lambda_{\text{aux}}^a} \int_E |u_a(z)|^2 \kappa_{\text{opt}}^a(dz) \\ &= \sum_{a=1}^N \frac{|w_{k-1}^a|^2}{\lambda_{\text{aux}}^a} |u_a^{\bullet\bullet}|^2, \end{aligned}$$

and it follows from Lemma 2.3 that the minimum w.r.t. the auxiliary probability vector is achieved for

$$\lambda_{\text{opt}}^a \propto w_{k-1}^a u_a^{\bullet\bullet},$$

and

$$\begin{aligned} \min_{(\lambda_{\text{aux}}, \kappa_{\text{aux}})} \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle &= \min_{\lambda_{\text{aux}}} \min_{\kappa_{\text{aux}}} \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle \\ &= \min_{\lambda_{\text{aux}}} \sum_{a=1}^N \frac{|w_{k-1}^a u_a^{\bullet\bullet}|^2}{\lambda_{\text{aux}}^a} \\ &= \left[\sum_{a=1}^N w_{k-1}^a u_a^{\bullet\bullet} \right]^2. \quad \square \end{aligned}$$

Remark 4.4 Recall that $u_a^{\bullet\bullet}$ denotes the minimum value of $u_a(z)$ when $z \in E$, while $u_a^\bullet$ denotes the minimum value of $u_a^c = u_a(\xi_{k-1}^c)$ when $c \in \{1, \dots, N\}$, hence $u_a^{\bullet\bullet} \leq u_a^\bullet$ for any $a \in \{1, \dots, N\}$ and

$$\left[\sum_{a=1}^N w_{k-1}^a u_a^{\bullet\bullet} \right]^2 \leq \left[\sum_{a=1}^N w_{k-1}^a u_a^\bullet \right]^2.$$

In other words, the optimal design for the particle filter with an auxiliary transition kernel improves the performance (reduces the variance) over the optimal design for the particle filter with an auxiliary transition matrix, and a fortiori it improves the performance over the optimal design for the auxiliary particle filter and over the ordinary (non auxiliary) particle filter.

Under the optimal design, independently for any $i \in \{1, \dots, N\}$

- the index a_k^i is sampled from the optimal probability vector $(\lambda_{\text{opt}}^1, \dots, \lambda_{\text{opt}}^N)$,

then, setting $a = a_k^i$ and $z = z_a^{\bullet\bullet}$

- the particle ξ_k^i is sampled from the probability distribution $Q_k(z, dx')$,

- and it receives the weight $w_k^i \propto g_{\text{opt}}^a(z, \xi_k^i)$, i.e.

$$w_k^i \propto g_k(\xi_k^i) \frac{q_k(\xi_{k-1}^a, \xi_k^i)}{q_k(z, \xi_k^i)} \frac{w_{k-1}^a}{\lambda_{\text{opt}}^a} \propto \frac{1}{u_a^{\bullet\bullet}} g_k(\xi_k^i) \frac{q_k(\xi_{k-1}^a, \xi_k^i)}{q_k(z, \xi_k^i)},$$

see Algorithm 6.

Algorithm 6: Particle filter with optimal auxiliary transition kernel

input : $(\xi_{k-1}^1, \dots, \xi_{k-1}^N), (w_{k-1}^1, \dots, w_{k-1}^N), Y_k$
output: $(\xi_k^1, \dots, \xi_k^N), (w_k^1, \dots, w_k^N)$
for $a = 1 \dots N$ **do**
 Optimize the design parameters
 Compute the minimum value $u_a^{\bullet\bullet}$ and the minimizer $z_a^{\bullet\bullet}$
 Evaluate the unnormalized optimal auxiliary weights
 Set $\lambda_{\text{opt}}^a = w_{k-1}^a u_a^{\bullet\bullet}$
end
Normalize the optimal auxiliary weights
for $i = 1 \dots N$ **do**
 Sample the auxiliary variable
 Sample a from the auxiliary probability vector $\lambda_{\text{opt}} = (\lambda_{\text{opt}}^1, \dots, \lambda_{\text{opt}}^N)$
 Propagate the state vector
 Set $z = z_a^{\bullet\bullet}$ and sample $\xi_k^i \sim Q_k(z, dx')$
 Evaluate the unnormalized weight
 Set $w_k^i = \frac{1}{u_a^{\bullet\bullet}} g_k(\xi_k^i) \frac{q_k(\xi_{k-1}^a, \xi_k^i)}{q_k(z, \xi_k^i)}$
end
Normalize the weights

To implement this optimal design, the challenge is to compute

$$u_a(z) = \left\{ \int_E |g_k(x') \frac{q_k(\xi_{k-1}^a, x')}{q_k(z, x')}|^2 Q_k(z, dx') \right\}^{1/2},$$

which was defined in (7), for any $a \in \{1, \dots, N\}$ and any $z \in E$, so as to compute the minimum value $u_a^{\bullet\bullet}$, find the minimizer $z_a^{\bullet\bullet}$ and set the optimal weight λ_{opt}^a , for any $a \in \{1, \dots, N\}$. As noticed earlier, it is not really necessary to compute $u_a(z)$ for any $z \in E$, it is only necessary to compute the minimum value and find the minimizer, which can be achieved after a few iterations of a numerical optimization procedure. This is illustrated in Section 6 in the special case where the hidden state is modelled as a linear Gaussian system, a special case of practical interest in applications.

5 Application to crossover and to multitarget tracking

It is well known that sequential Monte Carlo methods can be interpreted in terms of implementing *selection* and *mutation* steps, using the language of evolutionary algorithms. However, most

general evolutionary algorithms also include a *crossover* step that is not present in sequential Monte Carlo methods. The approach presented in the previous two sections can be adapted to include such a crossover step.

Assume that the state space $E = E_1 \times \dots \times E_T$ is decomposed as a product space. For instance in multitarget tracking, each different subspace would be the state space for a different target, and the multitarget state variable $x = (x_1, \dots, x_T)$ would be the concatenation of individual target state variables, with $x_t \in E_t$ the state variable of the t -th target for any $t = 1, \dots, T$.

The approach presented in Section 3, that uses an auxiliary transition matrix, can be adapted as follows (conceptually, the only difference is that the index c should be replaced by the multi-index $(c_1, \dots, c_T)$). Introducing the auxiliary $N \times (N \times \dots \times N)$ transition probability matrix $\pi_{\text{aux}} = (\pi_{\text{aux}}^{i, (j_1, \dots, j_T)})$ from $\{1, \dots, N\}$ to $\{1, \dots, N\} \times \dots \times \{1, \dots, N\}$, equivalently seen as a collection indexed by $i \in \{1, \dots, N\}$ of probability vectors on $\{1, \dots, N\} \times \dots \times \{1, \dots, N\}$, the idea is to see $\mu(dx')$ as the marginal probability distribution on E of the joint probability distribution

$$\mu_{\text{aux}}^{a, (c_1, \dots, c_T)}(dx') \propto g_k(x') w_{k-1}^a \pi_{\text{aux}}^{a, (c_1, \dots, c_T)} Q_k(\xi_{k-1}^a, dx') ,$$

defined on the augmented space $\{1, \dots, N\} \times \{1, \dots, N\} \times \dots \times \{1, \dots, N\} \times E$. Indeed, summation with respect to the index $a \in \{1, \dots, N\}$ and to the multi-index $(c_1, \dots, c_T) \in \{1, \dots, N\} \times \dots \times \{1, \dots, N\}$, yields

$$g_k(x') \sum_{a=1}^N w_{k-1}^a \left[\sum_{c_1=1}^N \dots \sum_{c_T=1}^N \pi_{\text{aux}}^{a, (c_1, \dots, c_T)} \right] Q_k(\xi_{k-1}^a, dx') = g(x') \eta(dx') , \quad (10)$$

an identity for unnormalized distributions, that carries over to normalized probability distributions.

Definition 5.1 For any multi-index $(c_1, \dots, c_T) \in \{1, \dots, N\} \times \dots \times \{1, \dots, N\}$, the shuffled multitarget particle $\xi_{k-1}^{(c_1, \dots, c_T)} = (\xi_{k-1,1}^{c_1}, \dots, \xi_{k-1,T}^{c_T})$ is obtained by taking its first component from the c_1 -th particle, its second component from the c_2 -th particle, and so forth up to taking its last component from the c_T -th particle.

Assuming that the model transition kernels have a density, i.e. assuming that

$$Q_k(x, dx') = q_k(x, x') dx' ,$$

makes it possible to define the importance decomposition

$$\mu_{\text{aux}}^{a, (c_1, \dots, c_T)}(dx') \propto g_k(x') \underbrace{\frac{q_k(\xi_{k-1}^a, x')}{q_k(\xi_{k-1}^{(c_1, \dots, c_T)}, x')} \frac{w_{k-1}^a}{\lambda_{\text{aux}}^a}}_{g_{\text{aux}}^{a, (c_1, \dots, c_T)}(x')} \underbrace{\lambda_{\text{aux}}^a \pi_{\text{aux}}^{a, (c_1, \dots, c_T)} Q_k(\xi_{k-1}^{(c_1, \dots, c_T)}, dx')}_{\eta_{\text{aux}}^{a, (c_1, \dots, c_T)}(dx')} , \quad (11)$$

the Monte Carlo approximation for the predictor

$$\eta_{\text{aux}}^N = \frac{1}{N} \sum_{i=1}^N \delta(a_k^i, c_{k,1}^i, \dots, c_{k,T}^i, \xi_k^i) \quad \text{hence} \quad \langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle = \frac{1}{N} \sum_{i=1}^N g_{\text{aux}}^{a_k^i, (c_{k,1}^i, \dots, c_{k,T}^i)}(\xi_k^i) ,$$

and the importance sampling approximation for the filter

$$\mu_{\text{aux}}^N = \sum_{i=1}^N w_k^i \delta_{(a_k^i, c_{k,1}^i, \dots, c_{k,T}^i, \xi_k^i)} \quad \text{and its marginal} \quad \mu^N = \sum_{i=1}^N w_k^i \delta_{\xi_k^i},$$

where independently for any $i = 1, \dots, N$ the i -th particle $(a_k^i, c_{k,1}^i, \dots, c_{k,T}^i, \xi_k^i)$ is sampled from $\eta_{\text{aux}}^{a, (c_1, \dots, c_T)}(dx')$ and is assigned a weight w_k^i using $g_{\text{aux}}^{a, (c_1, \dots, c_T)}(x')$. In practice, independently for any $i = 1, \dots, N$

- the index a_k^i is sampled from the auxiliary probability vector $(\lambda_{\text{aux}}^1, \dots, \lambda_{\text{aux}}^N)$,

then, setting $a = a_k^i$

- the multi-index $(c_{k,1}^i, \dots, c_{k,T}^i)$ is sampled from the $N \times \dots \times N$ -dimensional joint probability vector $(\pi_{\text{aux}}^{a, (1, \dots, 1)}, \dots, \pi_{\text{aux}}^{a, (N, \dots, N)})$, the a -th row of the auxiliary transition matrix π_{aux} ,

then, setting $(c_1, \dots, c_T) = (c_{k,1}^i, \dots, c_{k,T}^i)$

- the particle ξ_k^i is sampled from the probability distribution $Q_k(\xi_{k-1}^{(c_1, \dots, c_T)}, dx')$,
- and it receives the weight $w_k^i \propto g_{\text{aux}}^{a, (c_1, \dots, c_T)}(\xi_k^i)$, i.e.

$$w_k^i \propto g_k(\xi_k^i) \frac{q_k(\xi_{k-1}^a, \xi_k^i)}{q_k(\xi_{k-1}^{(c_1, \dots, c_T)}, \xi_k^i)} \frac{w_{k-1}^a}{\lambda_{\text{aux}}^a},$$

see Algorithm 7.

Remark 5.2 With this approach, multitarget particles such as $\xi_{k-1}^a = (\xi_{k-1,1}^a, \dots, \xi_{k-1,T}^a)$ that are present in the population initially, can be replaced by shuffled multitarget particles such as $\xi_{k-1}^{(c_1, \dots, c_T)} = (\xi_{k-1,1}^{c_1}, \dots, \xi_{k-1,T}^{c_T})$. It is the role of the auxiliary transition matrix π_{aux} to make this crossover between the target particles possible.

This class of particle filters depends upon $(N-1) + N(N^T-1) = N^{T+1}-1$ design parameters, that are the components of the auxiliary probability vector λ_{aux} (a probability vector of dimension N) and the components of the N rows of the auxiliary transition matrix π_{aux} (N probability vectors of dimension N^T each), subject to normalization constraints. This class of particle filters contains the class of auxiliary particle filters as a special case. Indeed

- if for any $a \in \{1, \dots, N\}$, the a -th row of the auxiliary transition matrix π_{aux} satisfies $\pi_{\text{aux}}^{a, (c_1, \dots, c_T)} = 1$ if $c_1 = \dots = c_T = a$ and $\pi_{\text{aux}}^{a, (c_1, \dots, c_T)} = 0$ otherwise, then the resulting particle filter belongs to the class of auxiliary particle filters described in Section 2.

Algorithm 7: Multitarget particle filter with auxiliary transition matrix

input : $\lambda_{\text{aux}}, \pi_{\text{aux}}, (\xi_{k-1}^1, \dots, \xi_{k-1}^N), (w_{k-1}^1, \dots, w_{k-1}^N), Y_k$
output: $(\xi_k^1, \dots, \xi_k^N), (w_k^1, \dots, w_k^N)$
for $i = 1$ *to* N **do**
 Sample the auxiliary variable
 Sample a from the auxiliary probability vector $\lambda_{\text{aux}} = (\lambda_{\text{aux}}^1, \dots, \lambda_{\text{aux}}^N)$
 Sample the shuffling multi-index
 Sample $(c_1, \dots, c_T)$ from the a -th row of the auxiliary transition matrix π_{aux} , i.e.
 from the probability vector $(\pi_{\text{aux}}^{a,(1,\dots,1)}, \dots, \pi_{\text{aux}}^{a,(N,\dots,N)})$
 Propagate the multitarget state vector
 Sample $\xi_k^i \sim Q_k(\xi_{k-1}^{(c_1, \dots, c_T)}, dx')$
 Evaluate the unnormalized weights
 Set $w_k^i = g_k(\xi_k^i) \frac{w_{k-1}^a}{\lambda_{\text{aux}}^a} \frac{q_{k,t}(\xi_{k-1}^a, \xi_k^i)}{q_{k,t}(\xi_{k-1}^{(c_1, \dots, c_T)}, \xi_k^i)}$
end
 Normalize the weights

Recall the particle approximation

$$\langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle = \frac{1}{N} \sum_{i=1}^N g_{\text{aux}}^{a_k^i, (c_{k,1}^i, \dots, c_{k,T}^i)}(\xi_k^i),$$

for the normalizing constant $\langle \eta_{\text{aux}}, g_{\text{aux}} \rangle$, where the random variables $(a_k^i, c_{k,1}^i, \dots, c_{k,T}^i, \xi_k^i)$ for $i = 1, \dots, N$ are i.i.d. with common probability distribution $\eta_{\text{aux}}^{a,(c_1, \dots, c_T)}(dx')$, hence

$$\mathbb{E}[\langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle] = \langle \eta_{\text{aux}}, g_{\text{aux}} \rangle,$$

i.e. the approximation is unbiased, and

$$N \mathbb{E} |\langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle - \langle \eta_{\text{aux}}, g_{\text{aux}} \rangle|^2 = \text{var}(g_{\text{aux}}, \eta_{\text{aux}}) = \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle - \langle \eta_{\text{aux}}, g_{\text{aux}} \rangle^2.$$

Remark 5.3 Using the identity (10) and the importance decomposition (11) yields

$$\sum_{a=1}^N \sum_{c_1=1}^N \dots \sum_{c_T=1}^N g_{\text{aux}}^{a,(c_1, \dots, c_T)}(x') \eta_{\text{aux}}^{a,(c_1, \dots, c_T)}(dx') = g(x') \eta(dx'),$$

an identity for unnormalized distributions, that implies identity for normalizing constants. Indeed, integration of both sides with respect to the variable $x' \in E$ shows that the normalizing constant $\langle \eta_{\text{aux}}, g_{\text{aux}} \rangle$ associated with the importance decomposition (11) coincides with the normalizing constant $\langle \eta, g \rangle$.

Remark 5.4 Actually, $\langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle$ provides also an unbiased approximation of the normalizing constant $\langle \eta, g \rangle$, and the variance of the approximation error satisfies

$$N \mathbb{E} |\langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle - \langle \eta, g \rangle|^2 = \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle - \langle \eta, g \rangle^2,$$

therefore minimizing this expression w.r.t. the auxiliary weights and the auxiliary transition matrix, reduces to minimizing $\langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle$ w.r.t. the auxiliary weights and the auxiliary transition matrix. Clearly, the minimum value is smaller than the value obtained for any choice of the auxiliary weights and the auxiliary transition matrix, and in particular is smaller than the value obtained for the special choice corresponding to the ordinary (non auxiliary) particle filter, hence

$$\min_{(\lambda_{\text{aux}}, \pi_{\text{aux}})} \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle \leq \langle \eta, g^2 \rangle \quad \text{and} \quad \min_{(\lambda_{\text{aux}}, \pi_{\text{aux}})} \text{var}(g_{\text{aux}}, \eta_{\text{aux}}) \leq \text{var}(g, \eta) .$$

Let

$$u_a^{(c_1, \dots, c_T)} = \left\{ \int_E |g_k(x') \frac{q_k(\xi_{k-1}^a, x')}{q_k(\xi_{k-1}^{(c_1, \dots, c_T)}, x')}|^2 Q_k(\xi_{k-1}^{(c_1, \dots, c_T)}, dx') \right\}^{1/2} , \quad (12)$$

and note that in the special case where $c_1 = \dots = c_T = a$, this expression reduces to

$$u_a^{(a, \dots, a)} = \left\{ \int_E |g_k(x')|^2 Q_k(\xi_{k-1}^a, dx') \right\}^{1/2} = u_a ,$$

which was defined in (3). The optimal design, that minimizes the variance of $\langle \eta_{\text{aux}}^N, g_{\text{aux}} \rangle$ seen as an unbiased approximation of the normalizing constant $\langle \eta, g \rangle$, is defined as follows.

Proposition 5.5 *For any $a \in \{1 \dots N\}$, let $u_a^\bullet$ denote the minimum value of $u_a^{(c_1, \dots, c_T)}$ when $(c_1, \dots, c_T) \in \{1 \dots N\} \times \dots \times \{1 \dots N\}$ and assume that there exists a unique minimizer, i.e. a unique multi-index $(c_{a,1}^\bullet, \dots, c_{a,T}^\bullet)$ for which the minimum value is achieved. Then for any $a \in \{1 \dots N\}$ the optimal choice for the a -th row of the auxiliary transition matrix is such that*

$$\pi_{\text{opt}}^{a, (c_1, \dots, c_T)} = 1 \text{ if } (c_1, \dots, c_T) = (c_{a,1}^\bullet, \dots, c_{a,T}^\bullet) \text{ and } \pi_{\text{opt}}^{a, (c_1, \dots, c_T)} = 0 \text{ otherwise,}$$

the optimal choice for the auxiliary weight is

$$\lambda_{\text{opt}}^a \propto w_{k-1}^a u_a^\bullet ,$$

and the minimum value is

$$\min_{(\lambda_{\text{aux}}, \pi_{\text{aux}})} \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle = \left[\sum_{a=1}^N w_{k-1}^a u_a^\bullet \right]^2 .$$

PROOF. The variance of the approximation error is controlled by

$$\begin{aligned} \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle &= \sum_{a=1}^N \sum_{c_1=1}^N \dots \sum_{c_T=1}^N \int_E |g_{\text{aux}}^{a, (c_1, \dots, c_T)}(x')|^2 \eta_{\text{aux}}^{a, (c_1, \dots, c_T)}(dx') \\ &= \sum_{a=1}^N \sum_{c_1=1}^N \dots \sum_{c_T=1}^N \int_E |g_k(x') \frac{q_k(\xi_{k-1}^a, x')}{q_k(\xi_{k-1}^{(c_1, \dots, c_T)}, x')} \frac{w_{k-1}^a}{\lambda_{\text{aux}}^a}|^2 \\ &\quad \lambda_{\text{aux}}^a \pi_{\text{aux}}^{a, (c_1, \dots, c_T)} Q_k(\xi_{k-1}^{(c_1, \dots, c_T)}, dx') \end{aligned}$$

$$\begin{aligned}
&= \sum_{a=1}^N \frac{|w_{k-1}^a|^2}{\lambda_{\text{aux}}^a} \sum_{c_1=1}^N \cdots \sum_{c_T=1}^N \pi_{\text{aux}}^{a,(c_1,\dots,c_T)} \\
&\quad \int_E |g_k(x') \frac{q_k(\xi_{k-1}^a, x')}{q_k(\xi_{k-1}^{(c_1,\dots,c_T)}, x')}|^2 Q_k(\xi_{k-1}^{(c_1,\dots,c_T)}, dx') \\
&= \sum_{a=1}^N \frac{|w_{k-1}^a|^2}{\lambda_{\text{aux}}^a} \sum_{c_1=1}^N \cdots \sum_{c_T=1}^N \pi_{\text{aux}}^{a,(c_1,\dots,c_T)} |u_a^{(c_1,\dots,c_T)}|^2 .
\end{aligned}$$

Under the assumptions, it holds

$$\sum_{c_1=1}^N \cdots \sum_{c_T=1}^N \pi_{\text{aux}}^{a,(c_1,\dots,c_T)} |u_a^{(c_1,\dots,c_T)}|^2 \geq |u_a^\bullet|^2 ,$$

and the lower bound is achieved if the a -th row of the $N \times (N \times \cdots \times N)$ transition matrix π_{aux} charges the minimizer only, i.e. if it satisfies $\pi_{\text{aux}}^{a,(c_1,\dots,c_T)} = 1$ if $(c_1, \dots, c_T) = (c_{a,1}^\bullet, \dots, c_{a,T}^\bullet)$ and $\pi_{\text{aux}}^{a,(c_1,\dots,c_T)} = 0$ otherwise. Plugging this expression yields

$$\begin{aligned}
\min_{\pi_{\text{aux}}} \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle &= \min_{\pi_{\text{aux}}} \sum_{a=1}^N \frac{|w_{k-1}^a|^2}{\lambda_{\text{aux}}^a} \sum_{c_1=1}^N \cdots \sum_{c_T=1}^N \pi_{\text{aux}}^{a,(c_1,\dots,c_T)} |u_a^{(c_1,\dots,c_T)}|^2 \\
&= \sum_{a=1}^N \frac{|w_{k-1}^a|^2}{\lambda_{\text{aux}}^a} \sum_{c_1=1}^N \cdots \sum_{c_T=1}^N \pi_{\text{opt}}^{a,(c_1,\dots,c_T)} |u_a^{(c_1,\dots,c_T)}|^2 \\
&= \sum_{a=1}^N \frac{|w_{k-1}^a|^2}{\lambda_{\text{aux}}^a} |u_a^\bullet|^2 ,
\end{aligned}$$

and it follows from Lemma 2.3 that the minimum w.r.t. the auxiliary weights is achieved for

$$\lambda_{\text{opt}}^a \propto w_{k-1}^a u_a^\bullet ,$$

and

$$\begin{aligned}
\min_{(\lambda_{\text{aux}}, \pi_{\text{aux}})} \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle &= \min_{\lambda_{\text{aux}}} \min_{\pi_{\text{aux}}} \langle \eta_{\text{aux}}, g_{\text{aux}}^2 \rangle \\
&= \min_{\lambda_{\text{aux}}} \sum_{a=1}^N \frac{|w_{k-1}^a u_a^\bullet|^2}{\lambda_{\text{aux}}^a} \\
&= \left[\sum_{a=1}^N w_{k-1}^a u_a^\bullet \right]^2 . \quad \square
\end{aligned}$$

Remark 5.6 Recall that $u_a^\bullet$ denotes the minimum value of $u_a^{(c_1,\dots,c_T)}$ when $(c_1, \dots, c_T) \in \{1 \cdots N\} \times \cdots \times \{1 \cdots N\}$ and in particular $u_a^{(c_1,\dots,c_T)} \leq u_a^{(a,\dots,a)} = u_a$, hence

$$\left[\sum_{a=1}^N w_{k-1}^a u_a^\bullet \right]^2 \leq \left[\sum_{a=1}^N w_{k-1}^a u_a \right]^2 .$$

In other words, the optimal design using an additional crossover step in the auxiliary particle filter improves the performance (reduces the variance) over the optimal design for the auxiliary particle filter.

Under the optimal design, independently for any $i \in \{1, \dots, N\}$

- the index a_k^i is sampled from the optimal probability vector $(\lambda_{\text{opt}}^1, \dots, \lambda_{\text{opt}}^N)$,

then, setting $a = a_k^i$ and $(c_1, \dots, c_T) = (c_{a,1}^\bullet, \dots, c_{a,T}^\bullet)$

- the particle ξ_k^i is sampled from the probability distribution $Q_k(\xi_{k-1}^{(c_1, \dots, c_T)}, dx')$,
- and it receives the weight $w_k^i \propto g_{\text{opt}}^{a, (c_1, \dots, c_T)}(\xi_k^i)$, i.e.

$$w_k^i \propto g_k(\xi_k^i) \frac{q_k(\xi_{k-1}^a, \xi_k^i)}{q_k(\xi_{k-1}^{(c_1, \dots, c_T)}, \xi_k^i)} \frac{w_{k-1}^a}{\lambda_{\text{opt}}^a} \propto \frac{1}{u_a^\bullet} g_k(\xi_k^i) \frac{q_k(\xi_{k-1}^a, \xi_k^i)}{q_k(\xi_{k-1}^{(c_1, \dots, c_T)}, \xi_k^i)},$$

see Algorithm 8.

Algorithm 8: Multitarget particle filter with optimal auxiliary transition matrix

input : $(\xi_{k-1}^1, \dots, \xi_{k-1}^N), (w_{k-1}^1, \dots, w_{k-1}^N), Y_k$

output: $(\xi_k^1, \dots, \xi_k^N), (w_k^1, \dots, w_k^N)$

for $a = 1 \dots N$ **do**

 Optimize the design parameters

 Compute the minimum value $u_a^\bullet$ and the minimizer $c_a^\bullet = (c_{a,1}^\bullet, \dots, c_{a,T}^\bullet)$

 Evaluate the unnormalized optimal weights

 Set $\lambda_{\text{opt}}^a = w_{k-1}^a u_a^\bullet$

end

Normalize the optimal weights

for $i = 1 \dots N$ **do**

 Sample the auxiliary variable

 Sample a from the optimal probability vector $\lambda_{\text{opt}} = (\lambda_{\text{opt}}^1, \dots, \lambda_{\text{opt}}^N)$

 Propagate the multitarget state vector

 Set $(c_1, \dots, c_T) = (c_{a,1}^\bullet, \dots, c_{a,T}^\bullet)$ and sample $\xi_k^i \sim Q_k(\xi_{k-1}^{(c_1, \dots, c_T)}, dx')$

 Evaluate the unnormalized weight

 Set $w_k^i = \frac{1}{u_a^\bullet} g_k(\xi_k^i) \frac{q_k(\xi_{k-1}^a, \xi_k^i)}{q_k(\xi_{k-1}^{(c_1, \dots, c_T)}, \xi_k^i)}$

end

Normalize the weights

To implement this optimal design, the challenge is to compute

$$u_a^{(c_1, \dots, c_T)} = \left\{ \int_E |g_k(x') \frac{q_k(\xi_{k-1}^a, x')}{q_k(\xi_{k-1}^{(c_1, \dots, c_T)}, x')}|^2 Q_k(\xi_{k-1}^{(c_1, \dots, c_T)}, dx') \right\}^{1/2},$$

which was defined in (12), for any index $a \in \{1, \dots, N\}$ and any multi-index $(c_1, \dots, c_T) \in \{1, \dots, N\} \times \dots \times \{1, \dots, N\}$, so as to compute the minimum value $u_a^\bullet$, find the minimizer $(c_{a,1}^\bullet, \dots, c_{a,T}^\bullet)$ and set the optimal weight λ_{opt}^a , for any $a \in \{1, \dots, N\}$. Recall the expression of the mapping

$$u_a(z) = \left\{ \int_E |g_k(x') \frac{q_k(\xi_{k-1}^a, x')}{q_k(z, x')}|^2 Q_k(z, dx') \right\}^{1/2},$$

which was defined in (7) and note that in the special case where $z = \xi_{k-1}^{(c_1, \dots, c_T)}$ this expression reduces to

$$u_a(\xi_{k-1}^{(c_1, \dots, c_T)}) = \left\{ \int_E |g_k(x') \frac{q_k(\xi_{k-1}^a, x')}{q_k(\xi_{k-1}^{(c_1, \dots, c_T)}, x')}|^2 Q_k(\xi_{k-1}^{(c_1, \dots, c_T)}, dx') \right\}^{1/2} = u_a^{(c_1, \dots, c_T)}.$$

Remark 5.7 The problem of minimizing $u_a^{(c_1, \dots, c_T)} = u_a(\xi_{k-1}^{(c_1, \dots, c_T)})$ with respect to the multi-index $(c_1, \dots, c_T) \in \{1, \dots, N\} \times \dots \times \{1, \dots, N\}$ considers all possible shuffled multitarget particles and can only be solved by exhaustive search within a finite (but huge) set. If this combinatorial minimization problem would be replaced by the wider problem of minimizing $u_a(z)$ with respect to the variable $z \in E$, as if there would be a continuum of possible choices and not a finite (but huge) number $N \times \dots \times N$ (T -times) of possible choices, then it would be possible to use numerical optimization procedures instead of exhaustive search.

The larger class of particle filters already considered in Section 4, with an auxiliary transition kernel, would make this possible and could be implemented as such, virtually without any adaptation.

Remark 5.8 With the approach presented in Section 4, multitarget particles such as $\xi_{k-1}^a = (\xi_{k-1,1}^a, \dots, \xi_{k-1,T}^a)$ that are present in the population initially, can be replaced by practically *any* multitarget particles, not limited to shuffled multitarget particles such as $\xi_{k-1}^{(c_1, \dots, c_T)} = (\xi_{k-1,1}^{c_1}, \dots, \xi_{k-1,T}^{c_T})$. It is the role of the auxiliary transition kernel κ_{aux} to make this proposition of new multitarget particles possible.

This class of particle filters depends upon several design parameters, that are the components of the auxiliary probability vector λ_{aux} (a probability vector of dimension N) subject to normalization constraint and the N 'rows' of the auxiliary transition kernel κ_{aux} (N probability distributions defined on E each). This class contains the class of particle filters defined above, with an auxiliary transition matrix — and a fortiori it contains the class of auxiliary particle filters — as a special case. Indeed

- if for any $a \in \{1, \dots, N\}$ the a -th 'row' of auxiliary transition kernel κ_{aux} charges the finite set of all possible shuffled multitarget particles $\xi_{k-1}^{(c_1, \dots, c_T)}$ only, i.e. if

$$\kappa_{\text{aux}}^a(dz) = \sum_{c_1=1}^N \dots \sum_{c_T=1}^N \pi_{\text{aux}}^{a, (c_1, \dots, c_T)} \delta_{\xi_{k-1}^{(c_1, \dots, c_T)}}(dz),$$

as a finite mixture, where the vector of mixture weights is given as the a -th row of some transition matrix π_{aux} of dimension $N \times (N \times \dots \times N)$, then the resulting particle filter belongs to the class of particle filters with an auxiliary transition matrix described at the beginning of this section.

6 Special case: linear Gaussian model transitions

Explicit calculations, and the proof of the existence of a *unique* minimizer for the mapping $u_a(z)$ defined in (7), are available in the special case where the hidden state is modelled as a linear Gaussian system

$$X_k = F_k X_{k-1} + W_k ,$$

where W_k is a Gaussian random vector with zero mean and invertible covariance matrix Σ_k , so that the model transition densities are Gaussian densities of the form

$$q_k(x, x') \propto \exp\{-\frac{1}{2} (x' - F_k x)^* \Sigma_k^{-1} (x' - F_k x)\} ,$$

for any $x, x' \in E$. For simplicity, the notation $m_a = \xi_{k-1}^a$ is used throughout this and the next sections.

Proposition 6.1 *For any $a \in \{1, \dots, N\}$, the mapping*

$$u_a(z) = \left\{ \int_E |g_k(x') \frac{q_k(m_a, x')}{q_k(z, x')}|^2 Q_k(z, dx') \right\}^{1/2}$$

which was defined in (7), can be expressed as $u_a(z) = v_a(F_k(m_a - z))$ in terms of the new variable $\theta = F_k(m_a - z)$ and the reduced mapping

$$v_a(\theta) = \exp\{\frac{1}{2} \theta^* \Sigma_k^{-1} \theta\} \left\{ \int_E |g_k(x'' + F_k m_a)|^2 \exp\{-\frac{1}{2} (x'' - \theta)^* \Sigma_k^{-1} (x'' - \theta)\} dx'' \right\}^{1/2} , \quad (13)$$

defined up to a multiplicative constant that does not depend on $\theta \in E$ nor on $a \in \{1, \dots, N\}$.

PROOF. For simplicity, write $F_k = F$ and $\Sigma_k = \Sigma$, i.e. the subscript index k is dropped throughout the proof. It follows from (the first part of) Lemma A.1 that

$$\begin{aligned} \frac{q_k(m_a, x')}{q_k(z, x')} &= \frac{\exp\{-\frac{1}{2} (x' - F m_a)^* \Sigma^{-1} (x' - F m_a)\}}{\exp\{-\frac{1}{2} (x' - F z)^* \Sigma^{-1} (x' - F z)\}} \\ &= \exp\{(m_a - z)^* F^* \Sigma^{-1} (x' - F m_a)\} \exp\{\frac{1}{2} (m_a - z)^* F^* \Sigma^{-1} F (m_a - z)\} , \end{aligned}$$

hence

$$\begin{aligned} g_{\text{aux}}^a(z, x') &= g_k(x') \frac{q_k(m_a, x')}{q_k(z, x')} \frac{w_{k-1}^a}{\lambda_{\text{aux}}^a} \\ &= g_k(x') \frac{w_{k-1}^a}{\lambda_{\text{aux}}^a} \exp\{(m_a - z)^* F^* \Sigma^{-1} (x' - F m_a)\} \\ &\quad \exp\{\frac{1}{2} (m_a - z)^* F^* \Sigma^{-1} F (m_a - z)\} , \end{aligned}$$

for the weight function. It follows from (the second part of) Lemma A.1 that

$$\begin{aligned}
& \left| \frac{q_k(m_a, x')}{q_k(z, x')} \right|^2 q_k(z, x') \\
&= \left| \frac{\exp\{-\frac{1}{2} (x' - F m_a)^* \Sigma^{-1} (x' - F m_a)\}}{\exp\{-\frac{1}{2} (x' - F z)^* \Sigma^{-1} (x' - F z)\}} \right|^2 \exp\{-\frac{1}{2} (x' - F z)^* \Sigma^{-1} (x' - F z)\} \\
&= \exp\{-\frac{1}{2} (x' - F (m_a + (m_a - z)))^* \Sigma^{-1} (x' - F (m_a + (m_a - z)))\} \\
&\quad \exp\{(m_a - z)^* F^* \Sigma^{-1} F (m_a - z)\} ,
\end{aligned}$$

hence integration with respect to the variable $x' \in E$ yields

$$\begin{aligned}
u_a(z) &= \left\{ \int_E |g_k(x')| \frac{q_k(m_a, x')}{q_k(z, x')} Q_k(z, dx') \right\}^{1/2} \\
&= \exp\{\frac{1}{2} (m_a - z)^* F^* \Sigma^{-1} F (m_a - z)\} \left\{ \int_E |g_k(x')|^2 \right. \\
&\quad \left. \exp\{-\frac{1}{2} (x' - F (m_a + (m_a - z)))^* \Sigma^{-1} (x' - F (m_a + (m_a - z)))\} dx' \right\}^{1/2} \\
&= \exp\{\frac{1}{2} (m_a - z)^* F^* \Sigma^{-1} F (m_a - z)\} \left\{ \int_E |g_k(x'' + F m_a)|^2 \right. \\
&\quad \left. \exp\{-\frac{1}{2} (x'' - F (m_a - z))^* \Sigma^{-1} (x'' - F (m_a - z))\} dx'' \right\}^{1/2} ,
\end{aligned}$$

up to a multiplicative constant that does not depend on $z \in E$ nor on $a \in \{1, \dots, N\}$. Notice that the model transition density $q_k(z, x')$ depends on z only through $F z$ and the two functions $g_{\text{aux}}^a(z, x')$ and $u_a(z)$ depends on z only through $F(m_a - z)$. Indeed, $u_a(z) = v_a(F(m_a - z))$ and in particular $u_a(m_a) = v_a(0)$ with

$$v_a(\theta) = \exp\{\frac{1}{2} \theta^* \Sigma^{-1} \theta\} \left\{ \int_E |g_k(x'' + F m_a)|^2 \exp\{-\frac{1}{2} (x'' - \theta)^* \Sigma^{-1} (x'' - \theta)\} dx'' \right\}^{1/2} ,$$

and

$$\int_E |u_a(z)|^2 \kappa_{\text{aux}}^a(dz) = \int_E |v_a(F(m_a - z))|^2 \kappa_{\text{aux}}^a(dz) = \int_{\text{range}(F)} |v_a(\theta)|^2 \kappa_{\text{aux}}^a \circ F_a^{-1}(d\theta) ,$$

where $\kappa_{\text{aux}}^a \circ F_a^{-1}(d\theta)$ denotes the image (or pushforward) probability measure of $\kappa_{\text{aux}}^a(dz)$ under the change of variable $\theta = F_a(z) = F(m_a - z)$. In other words, this is the probability distribution of the random variable $F_a(Z) = F(m_a - Z)$ where the random variable Z has probability distribution $\kappa_{\text{aux}}^a(dz)$. $\square$

It follows from Proposition A.2 that the mapping $\theta \mapsto v_a(\theta)$ is strongly log-convex (indeed, $(\log v_a)''(\theta) \geq c I$, where the positive constant c does not depend on $\theta \in E$ nor on $a \in \{1, \dots, N\}$) hence it has a unique minimizer, and so does the mapping $z \mapsto u_a(z) = v_a(F_k(m_a - z))$ provided that the square matrix F_k is invertible. However, if the square matrix F_k is not invertible, then

- the mapping $z \mapsto u_a(z) = v_a(F_k(m_a - z))$ is still log-convex but not strongly log-convex and it cannot have a unique minimizer (indeed, if z is a minimizer, then $z + z_0$ is another minimizer, for any z_0 in the null space of F_k),
- when restricted to the range space of F_k , the mapping $\theta \mapsto v_a(\theta)$ is strongly log-convex, hence it has a unique minimizer.

The minimum value of $u_a(z)$ when $z \in E$, coincides with the minimum value of $v_a(F_k(m_a - z))$ when $z \in E$, hence the minimum value $u_a^{\bullet\bullet}$ can be interpreted as the minimum value of $v_a(\theta)$ subject to $\theta \in \text{range}(F_k)$. Even though the minimizer of $u_a(z)$ when $z \in E$ is not necessarily unique, any minimizer z satisfies $F_k(m_a - z) = \theta_a^{\bullet\bullet}$ or equivalently $F_k z = F_k m_a - \theta_a^{\bullet\bullet}$, where $\theta_a^{\bullet\bullet}$ is the unique minimizer of $v_a(\theta)$ subject to $\theta \in \text{range}(F_k)$. Then, setting $\theta = \theta_a^{\bullet\bullet}$

- the probability distribution $Q_k(z, dx')$ is a Gaussian probability distribution with mean $F_k z = F_k m_a - \theta$ and covariance matrix Σ_k ,
- the particle ξ_k^i sampled from the probability distribution $Q_k(z, dx')$ can be defined as $\xi_k^i = F_k m_a - \theta + \Xi_k^i$, where Ξ_k^i is a Gaussian random vector with zero mean and covariance matrix Σ_k , hence $\xi_k^i - F_k m_a = -\theta + \Xi_k^i$,

and therefore

$$\begin{aligned} \frac{q_k(m_a, \xi_k^i)}{q_k(z, \xi_k^i)} &= \exp\{(m_a - z)^* F_k^* \Sigma_k^{-1} (\xi_k^i - F_k m_a)\} \exp\{\frac{1}{2} (m_a - z)^* F_k^* \Sigma_k^{-1} F_k (m_a - z)\} \\ &= \exp\{\theta^* \Sigma_k^{-1} (-\theta + \Xi_k^i)\} \exp\{\frac{1}{2} \theta^* \Sigma_k^{-1} \theta\} \\ &= \exp\{\theta^* \Sigma_k^{-1} \Xi_k^i - \frac{1}{2} \theta^* \Sigma_k^{-1} \theta\} . \end{aligned}$$

Under the optimal design, independently for any $i \in \{1, \dots, N\}$

- the index a_k^i is sampled from the optimal probability vector $(\lambda_{\text{opt}}^1, \dots, \lambda_{\text{opt}}^N)$,

then, setting $a = a_k^i$ and $\theta = \theta_a^{\bullet\bullet}$

- the particle ξ_k^i is defined as $\xi_k^i = F_k \xi_{k-1}^a - \theta + \Xi_k^i$ where Ξ_k^i is a Gaussian random vector with zero mean and covariance matrix Σ_k ,
- and it receives the weight $w_k^i \propto g_{\text{opt}}^a(z, \xi_k^i)$, i.e.

$$w_k^i \propto g_k(\xi_k^i) \frac{q_k(\xi_{k-1}^a, \xi_k^i)}{q_k(z, \xi_k^i)} \frac{w_{k-1}^a}{\lambda_{\text{opt}}^a} \propto \frac{1}{u_a^{\bullet\bullet}} g_k(\xi_k^i) \exp\{\theta^* \Sigma_k^{-1} \Xi_k^i - \frac{1}{2} \theta^* \Sigma_k^{-1} \theta\} ,$$

see Algorithm 9.

To implement this optimal design, the challenge is to compute

$$v_a(\theta) = \exp\{\frac{1}{2} \theta^* \Sigma_k^{-1} \theta\} \left\{ \int_E |g_k(x'' + F_k m_a)|^2 \exp\{-\frac{1}{2} (x'' - \theta)^* \Sigma_k^{-1} (x'' - \theta)\} dx'' \right\}^{1/2} ,$$

Algorithm 9: Particle filter with optimal transition kernel (Gaussian model)

input : $(\xi_{k-1}^1, \dots, \xi_{k-1}^N), (w_{k-1}^1, \dots, w_{k-1}^N), Y_k$
output: $(\xi_k^1, \dots, \xi_k^N), (w_k^1, \dots, w_k^N)$
for $a = 1 \dots N$ **do**
 Optimize the design parameters
 Compute the minimum value $u_a^{\bullet\bullet}$ and the minimizer $\theta_a^{\bullet\bullet}$
 Evaluate the unnormalized optimal auxiliary weights
 Set $\lambda_{\text{opt}}^a = w_{k-1}^a u_a^{\bullet\bullet}$
end
 Normalize the optimal auxiliary weights
for $i = 1 \dots N$ **do**
 Sample the auxiliary variable
 Sample a from the auxiliary probability vector $\lambda_{\text{opt}} = (\lambda_{\text{opt}}^1, \dots, \lambda_{\text{opt}}^N)$
 Propagate the state vector
 Set $\theta = \theta_a^{\bullet\bullet}$ and sample $\Xi \sim \mathcal{N}(0, \Sigma_k)$
 Set $\xi_k^i = (F_k \xi_{k-1}^a - \theta) + \Xi$
 Evaluate the unnormalized weight
 Set $w_k^i = \frac{1}{u_a^{\bullet\bullet}} g_k(\xi_k^i) \exp\{\theta^* \Sigma_k^{-1} \Xi - \frac{1}{2} \theta^* \Sigma_k^{-1} \theta\}$
end
 Normalize the weights

which was defined in (13), for any $a \in \{1, \dots, N\}$ and any $\theta \in \text{range}(F_k)$, so as to compute the minimum value $u_a^{\bullet\bullet}$, find the minimizer $\theta_a^{\bullet\bullet}$ and set the optimal weight λ_{opt}^a , for any $a \in \{1, \dots, N\}$. Finding the unique minimizer of the strongly convex mapping $\theta \mapsto \log v_a(\theta)$ is routinely transformed into finding the unique zero of the mapping $\theta \mapsto (\log v_a)'(\theta)$, and this can be achieved using a stochastic approximation algorithm, see Appendix B.

7 Special case: linear Gaussian system

The purpose of this section is to consider a simple enough special case where it is possible for any $a \in \{1, \dots, N\}$ to provide an explicit expression for the reduced mapping $v_a(\theta)$, and for the minimum value $u_a^{\bullet\bullet}$ and the unique minimizer $\theta_a^{\bullet\bullet}$. However this special case is of limited practical interest, since the Bayesian filter itself has an explicit expression in terms of the Kalman filter, and there is no need for a Monte Carlo numerical approximation in terms of particle filters.

Assume that the hidden state and the observations are modelled jointly as a linear Gaussian system

$$X_k = F_k X_{k-1} + W_k ,$$

$$Y_k = H_k X_k + V_k ,$$

where W_k and V_k are two independent Gaussian random vectors with zero mean and invertible covariance matrices Σ_k and R_k respectively, so that the model transition densities are Gaussian

densities of the form

$$q_k(x, x') \propto \exp\{-\frac{1}{2} (x' - F_k x)^* \Sigma_k^{-1} (x' - F_k x)\} ,$$

for any $x, x' \in E$, the emission densities are Gaussian densities of the form

$$\exp\{-\frac{1}{2} (y - H_k x')^* R_k^{-1} (y - H_k x')\} ,$$

for any $x' \in E$ and $y \in \mathbb{R}^d$, and the likelihood function is defined as

$$g_k(x') = \exp\{-\frac{1}{2} (Y_k - H_k x')^* R_k^{-1} (Y_k - H_k x')\} ,$$

for any $x' \in E$. For simplicity, the notations $m_a = \xi_{k-1}^a$ and $I_a = Y_k - H_k F_k m_a$ for innovation are used throughout this section.

Proposition 7.1 *For any $a \in \{1, \dots, N\}$, the mapping*

$$v_a(\theta) = \exp\{\frac{1}{2} \theta^* \Sigma_k^{-1} \theta\} \left\{ \int_E |g_k(x'' + F_k m_a)|^2 \exp\{-\frac{1}{2} (x'' - \theta)^* \Sigma_k^{-1} (x'' - \theta)\} dx'' \right\}^{1/2} ,$$

which was defined in (13), has the following explicit expression

$$v_a(\theta) = \exp\{\frac{1}{2} \theta^* \Sigma_k^{-1} \theta\} \exp\{-\frac{1}{4} (I_a - H_k \theta)^* [H_k \Sigma_k H_k^* + \frac{1}{2} R_k]^{-1} (I_a - H_k \theta)\} ,$$

up to a multiplicative constant that does not depend on $\theta \in E$ nor on $a \in \{1, \dots, N\}$. Moreover, the symmetric matrix

$$A = \Sigma_k^{-1} - \frac{1}{2} H_k^* [H_k \Sigma_k H_k^* + \frac{1}{2} R_k]^{-1} H_k ,$$

associated with the quadratic form $\log v_a(\theta)$, is positive definite.

Equivalently

$$\log v_a(\theta) = \frac{1}{2} \theta^* \Sigma_k^{-1} \theta - \frac{1}{4} (I_a - H_k \theta)^* [H_k \Sigma_k H_k^* + \frac{1}{2} R_k]^{-1} (I_a - H_k \theta) ,$$

and in the special case where $\theta = 0$, this expression reduces to

$$\log u_a = \log v_a(0) = -\frac{1}{4} I_a^* [H_k \Sigma_k H_k^* + \frac{1}{2} R_k]^{-1} I_a ,$$

up to an additive constant that does not depend on $\theta \in E$ nor on $a \in \{1, \dots, N\}$.

PROOF. For simplicity, write $F_k = F$ and $\Sigma_k = \Sigma$, and also $H_k = H$ and $R_k = R$, i.e. the subscript index k is dropped throughout the proof. It holds

$$\begin{aligned} g_k(x'' + F m_a) &= \exp\{-\frac{1}{2} (Y_k - H (x'' + F m_a))^* R^{-1} (Y_k - H (x'' + F m_a))\} \\ &= \exp\{-\frac{1}{2} (I_a - H x'')^* R^{-1} (I_a - H x'')\} , \end{aligned}$$

hence

$$\begin{aligned} &|g_k(x'' + F m_a)|^2 \exp\{-\frac{1}{2} (x'' - \theta)^* \Sigma^{-1} (x'' - \theta)\} \\ &= \exp\{-(I_a - H x'')^* R^{-1} (I_a - H x'')\} \exp\{-\frac{1}{2} (x'' - \theta)^* \Sigma^{-1} (x'' - \theta)\} \\ &= \exp\{-\frac{1}{2} (I_a - H x'')^* (\frac{1}{2} R)^{-1} (I_a - H x'')\} \exp\{-\frac{1}{2} (x'' - \theta)^* \Sigma^{-1} (x'' - \theta)\} . \end{aligned}$$

Notice that the expression

$$\exp\{-\frac{1}{2}(y - H x'')^* (\frac{1}{2} R)^{-1} (y - H x'')\} \exp\{-\frac{1}{2}(x'' - \theta)^* \Sigma^{-1} (x'' - \theta)\} , \quad (14)$$

can be interpreted (up to a normalizing constant) as the joint density of the random vector (X, Y) in the model $Y = H X + V$, where X and V are two independent Gaussian random vectors, with mean θ and 0 and with covariance matrix Σ and $\frac{1}{2} R$, respectively. Alternatively, this joint density can be factored as the product of the density of the random vector Y , a Gaussian density with mean $H \theta$ and with covariance matrix $H \Sigma H^* + \frac{1}{2} R$, and of the conditional density of the random vector X given $Y = y$, a Gaussian density with mean and covariance matrix

$$\hat{X}(y) = \theta + K (y - H \theta) \quad \text{and} \quad P = (I - K H) \Sigma ,$$

respectively, with the Kalman gain matrix

$$K = \Sigma H^* [H \Sigma H^* + \frac{1}{2} R]^{-1} .$$

Therefore, the expression (14) factors as

$$\begin{aligned} & \exp\{-\frac{1}{2}(y - H x'')^* (\frac{1}{2} R)^{-1} (y - H x'')\} \exp\{-\frac{1}{2}(x'' - \theta)^* \Sigma^{-1} (x'' - \theta)\} \\ &= \exp\{-\frac{1}{2}(y - H \theta)^* [H \Sigma H^* + \frac{1}{2} R]^{-1} (y - H \theta)\} \\ & \exp\{-\frac{1}{2}(x'' - \hat{X}(y))^* P^{-1} (x'' - \hat{X}(y))\} . \end{aligned} \quad (15)$$

Actually, this identity does not hold up to a multiplicative constant only. Indeed, keeping track of normalizing constants, the normalizing constant for the left-hand side should be

$$\sqrt{\det(2\pi \frac{1}{2} R)} \sqrt{\det(2\pi \Sigma)} ,$$

and the normalizing constant for the right-hand side should be

$$\sqrt{\det(2\pi (H \Sigma H^* + \frac{1}{2} R))} \sqrt{\det(2\pi P)} .$$

However, introducing the covariance block-matrix

$$\begin{pmatrix} \Sigma & \Sigma H^* \\ H \Sigma & H \Sigma H^* + \frac{1}{2} R \end{pmatrix} ,$$

considering that

$$\Sigma - \Sigma H^* [H \Sigma H^* + \frac{1}{2} R]^{-1} H \Sigma = P ,$$

is the Schur complement of $(H \Sigma H^* + \frac{1}{2} R)$ and that

$$(H \Sigma H^* + \frac{1}{2} R) - H \Sigma \Sigma^{-1} \Sigma H^* = \frac{1}{2} R ,$$

is the Schur complement of Σ , and using the Schur determinant formula, yields

$$\det(\frac{1}{2} R) \det \Sigma = \det(H \Sigma H^* + \frac{1}{2} R) \det P .$$

In other words, the missing normalizing constant for the left-hand side is equal to the missing normalizing constant for the right-hand side. The factorization (15) yields

$$\begin{aligned}
& |g_k(x'' + F m_a)|^2 \exp\{-\tfrac{1}{2} (x'' - \theta)^* \Sigma^{-1} (x'' - \theta)\} \\
&= \exp\{-\tfrac{1}{2} (I_a - H x'')^* (\tfrac{1}{2} R)^{-1} (I_a - H x'')\} \exp\{-\tfrac{1}{2} (x'' - \theta)^* \Sigma^{-1} (x'' - \theta)\} \\
&= \exp\{-\tfrac{1}{2} (I_a - H \theta)^* [H \Sigma H^* + \tfrac{1}{2} R]^{-1} (I_a - H \theta)\} \\
&\quad \exp\{-\tfrac{1}{2} (x'' - \hat{X}(I_a))^* P^{-1} (x'' - \hat{X}(I_a))\} .
\end{aligned}$$

Integration with respect to the variable x'' provides the normalizing constant, and normalization provides the expression for the normalized density $p(\theta, x)$, a Gaussian density with mean vector $\hat{X}(I_a)$ and with covariance matrix P . Indeed, keeping track of the normalizing constants

$$\begin{aligned}
& \int_E |g_k(x'' + F m_a)|^2 \exp\{-\tfrac{1}{2} (x'' - \theta)^* \Sigma^{-1} (x'' - \theta)\} \frac{dx''}{\sqrt{\det(2\pi\Sigma)}} \\
&= \frac{\sqrt{\det(2\pi P)}}{\sqrt{\det(2\pi\Sigma)}} \exp\{-\tfrac{1}{2} (I_a - H \theta)^* [H \Sigma H^* + \tfrac{1}{2} R]^{-1} (I_a - H \theta)\} \\
&\quad \int_E \exp\{-\tfrac{1}{2} (x'' - \hat{X}(I_a))^* P^{-1} (x'' - \hat{X}(I_a))\} \frac{dx''}{\sqrt{\det(2\pi P)}} \\
&= \frac{\sqrt{\det P}}{\sqrt{\det \Sigma}} \exp\{-\tfrac{1}{2} (I_a - H \theta)^* [H \Sigma H^* + \tfrac{1}{2} R]^{-1} (I_a - H \theta)\} ,
\end{aligned}$$

hence

$$v_a(\theta) = \left(\frac{\sqrt{\det P}}{\sqrt{\det \Sigma}}\right)^{1/2} \exp\{\tfrac{1}{2} \theta^* \Sigma^{-1} \theta\} \exp\{-\tfrac{1}{4} (I_a - H \theta)^* [H \Sigma H^* + \tfrac{1}{2} R]^{-1} (I_a - H \theta)\} ,$$

and

$$\log v_a(\theta) = \tfrac{1}{2} \theta^* \Sigma^{-1} \theta - \tfrac{1}{4} (I_a - H \theta)^* [H \Sigma H^* + \tfrac{1}{2} R]^{-1} (I_a - H \theta) + \tfrac{1}{4} \log \frac{\det P}{\det \Sigma} .$$

Note that the expression for $\log v_a(\theta)$ is the *difference* of two symmetric quadratic forms, a definite positive quadratic form and a semi-definite positive quadratic form, and this difference could in principle be any kind of a symmetric quadratic form. It follows from Proposition A.2 that the mapping $\theta \mapsto \log v_a(\theta)$ is strongly convex, whatever the emission density could be. This general result can be obtained directly here, i.e. it can be shown that the symmetric matrix

$$A = \Sigma^{-1} - \tfrac{1}{2} H^* [H \Sigma H^* + \tfrac{1}{2} R]^{-1} H ,$$

associated with the quadratic form $\log v_a(\theta)$, is positive definite. Indeed, using the matrix inversion lemma yields

$$[\Sigma + \Sigma H^* (\tfrac{1}{2} R)^{-1} H \Sigma]^{-1} = \Sigma^{-1} - H^* [H \Sigma H^* + \tfrac{1}{2} R]^{-1} H ,$$

hence

$$\begin{aligned}
A &= \Sigma^{-1} - \frac{1}{2} H^* [H \Sigma H^* + \frac{1}{2} R]^{-1} H \\
&= \frac{1}{2} \Sigma^{-1} + \frac{1}{2} \Sigma^{-1} - \frac{1}{2} H^* [H \Sigma H^* + \frac{1}{2} R]^{-1} H \\
&= \frac{1}{2} \Sigma^{-1} + \frac{1}{2} [\Sigma + \Sigma H^* (\frac{1}{2} R)^{-1} H \Sigma]^{-1} \geq \frac{1}{2} \Sigma^{-1} . \quad \square
\end{aligned}$$

Rewriting

$$\begin{aligned}
\log v_a(\theta) &= \frac{1}{2} \theta^* \Sigma_k^{-1} \theta - \frac{1}{4} (I_a - H_k \theta)^* [H_k \Sigma_k H_k^* + \frac{1}{2} R_k]^{-1} (I_a - H \theta) + \text{cst} \\
&= \frac{1}{2} \theta^* A \theta + b^* \theta + e \\
&= \frac{1}{2} (\theta + A^{-1} b)^* A (\theta + A^{-1} b) + e - \frac{1}{2} b^* A^{-1} b ,
\end{aligned}$$

where

$$b = \frac{1}{2} H_k^* [H_k \Sigma_k H_k^* + \frac{1}{2} R_k]^{-1} I_a ,$$

and

$$e = -\frac{1}{4} I_a^* [H_k \Sigma_k H_k^* + \frac{1}{2} R_k]^{-1} I_a + \text{cst} = \log u_a ,$$

it appears that there is an explicit expression for both the minimizer

$$\theta_a^{\bullet\bullet} = -A^{-1} b ,$$

and the minimum value

$$u_a^{\bullet\bullet} = \exp\{e - \frac{1}{2} b^* A^{-1} b\} = u_a \exp\{-\frac{1}{2} b^* A^{-1} b\} \leq u_a .$$

Note that

References

- [1] Yves F. Atchadé, Gersende Fort, and Éric Moulines. On perturbed proximal gradient algorithms. *Journal of Machine Learning Research*, 18(10):1–33, 2017.
- [2] Arkadii Nemirovskii, Anatoli Juditsky, Guanghui Lan, and Alexander Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on Optimization*, 19(4):1574–1609, 2009.
- [3] Yurii Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87 of *Applied Optimization*. Springer–Verlag, Boston, 2004.
- [4] Michael K. Pitt and Neil Shephard. Filtering via simulations: auxiliary particle filter. *Journal of the American Statistical Association*, 94(446):590–599, June 1999.

- [5] Luis Úbeda-Medina, Ángel F. García-Fernández, and Jesús Grajal. Adaptive auxiliary particle filter for track-before-detect with multiple targets. *IEEE Transactions on Aerospace and Electronic Systems*, AES-53(5):2317–2330, October 2017.
- [6] Wei Yi, Mark R. Morelande, Lingjiang Kong, and Jianyu Yang. A computationally efficient particle filter for multitarget tracking using an independence approximation. *IEEE Transactions on Signal Processing*, SP-61(4):843–856, February 2013.

A Gaussian computations

To begin with, here are some results on ratios of two Gaussian densities with different mean vectors m_0 and m , and the same invertible covariance matrix Σ .

Lemma A.1 *Let $\Delta = m_0 - m$ denote the difference of the two mean vectors. Then*

$$\begin{aligned} & \frac{\exp\{-\frac{1}{2}(x - m_0)^* \Sigma^{-1} (x - m_0)\}}{\exp\{-\frac{1}{2}(x - m)^* \Sigma^{-1} (x - m)\}} \\ &= \exp\{\frac{1}{2} \Delta^* \Sigma^{-1} \Delta\} \exp\{\Delta^* \Sigma^{-1} (x - m_0)\} , \end{aligned}$$

and

$$\begin{aligned} & \left| \frac{\exp\{-\frac{1}{2}(x - m_0)^* \Sigma^{-1} (x - m_0)\}}{\exp\{-\frac{1}{2}(x - m)^* \Sigma^{-1} (x - m)\}} \right|^2 \exp\{-\frac{1}{2}(x - m)^* \Sigma^{-1} (x - m)\} \\ &= \exp\{\Delta^* \Sigma^{-1} \Delta\} \exp\{-\frac{1}{2}(x - (m_0 + \Delta))^* \Sigma^{-1} (x - (m_0 + \Delta))\} . \end{aligned}$$

PROOF. Clearly

$$\begin{aligned} & \frac{\exp\{-\frac{1}{2}(x - m_0)^* \Sigma^{-1} (x - m_0)\}}{\exp\{-\frac{1}{2}(x - m)^* \Sigma^{-1} (x - m)\}} \\ &= \exp\{-\frac{1}{2}(x - m_0)^* \Sigma^{-1} (x - m_0) + \frac{1}{2}(x - m)^* \Sigma^{-1} (x - m)\} , \end{aligned}$$

and note that

$$\begin{aligned} \frac{1}{2}(x - m)^* \Sigma^{-1} (x - m) &= \frac{1}{2}(x - m_0 + \Delta)^* \Sigma^{-1} (x - m_0 + \Delta) \\ &= \frac{1}{2}(x - m_0)^* \Sigma^{-1} (x - m_0) + \Delta^* \Sigma^{-1} (x - m_0) + \frac{1}{2} \Delta^* \Sigma^{-1} \Delta , \end{aligned} \tag{16}$$

which shows the first part. Then

$$\begin{aligned} & \left| \frac{\exp\{-\frac{1}{2}(x - m_0)^* \Sigma^{-1} (x - m_0)\}}{\exp\{-\frac{1}{2}(x - m)^* \Sigma^{-1} (x - m)\}} \right|^2 \exp\{-\frac{1}{2}(x - m)^* \Sigma^{-1} (x - m)\} \\ &= \exp\{\Delta^* \Sigma^{-1} \Delta\} \exp\{2 \Delta^* \Sigma^{-1} (x - m_0)\} \exp\{-\frac{1}{2}(x - m)^* \Sigma^{-1} (x - m)\} , \end{aligned}$$

and using (16) yields

$$\begin{aligned}
& -\frac{1}{2} (x - m)^* \Sigma^{-1} (x - m) + 2 \Delta^* \Sigma^{-1} (x - m_0) \\
& = -\frac{1}{2} (x - m_0)^* \Sigma^{-1} (x - m_0) + \Delta^* \Sigma^{-1} (x - m_0) - \frac{1}{2} \Delta^* \Sigma^{-1} \Delta \\
& = -\frac{1}{2} (x - m_0 - \Delta)^* \Sigma^{-1} (x - m_0 - \Delta) ,
\end{aligned}$$

which shows the second part. $\square$

As a by-product, integration with respect to the variable x provides the expression for the χ^2 -divergence between two Gaussian densities with different mean vectors m_0 and m , and the same invertible covariance matrix Σ . Indeed

$$\begin{aligned}
& \int_E \left(\left| \frac{\exp\{-\frac{1}{2} (x - m_0)^* \Sigma^{-1} (x - m_0)\}}{\exp\{-\frac{1}{2} (x - m)^* \Sigma^{-1} (x - m)\}} \right|^2 - 1 \right) \exp\{-\frac{1}{2} (x - m)^* \Sigma^{-1} (x - m)\} \frac{dx}{\sqrt{\det(2\pi\Sigma)}} \\
& = \exp\{\Delta^* \Sigma^{-1} \Delta\} - 1 .
\end{aligned}$$

Note that the point $\mu = m_0 + \Delta = 2m_0 - m$ can be interpreted as the symmetric of the point m with respect to the central point m_0 , since (put in other words) the point $m_0 = \frac{1}{2} (m + \mu)$ is the middle of the segment joining the two points m and μ .

Proposition A.2 *For any function ϕ defined on E , the mapping*

$$\theta \mapsto \exp\{\frac{1}{2} \theta^* \Sigma^{-1} \theta\} \left\{ \int_E |\phi(x)|^2 \exp\{-\frac{1}{2} (x - \theta)^* \Sigma^{-1} (x - \theta)\} dx \right\}^{1/2} ,$$

defined up to a multiplicative constant that does not depend on $\theta \in E$, is strongly log-convex.

PROOF. For a given function ϕ defined on E , consider the two mappings

$$v(\theta) = \exp\{\frac{1}{2} \theta^* \Sigma^{-1} \theta\} \left\{ \int_E |\phi(x)|^2 \exp\{-\frac{1}{2} (x - \theta)^* \Sigma^{-1} (x - \theta)\} dx \right\}^{1/2} ,$$

and

$$v_{\text{int}}(\theta) = \int_E |\phi(x)|^2 \exp\{-\frac{1}{2} (x - \theta)^* \Sigma^{-1} (x - \theta)\} dx .$$

The first two derivatives (Jacobian row vector and Hessian symmetric matrix) are defined as

$$v'_{\text{int}}(\theta) = \int_E |\phi(x)|^2 (x - \theta)^* \Sigma^{-1} \exp\{-\frac{1}{2} (x - \theta)^* \Sigma^{-1} (x - \theta)\} dx ,$$

and

$$\begin{aligned}
v''_{\text{int}}(\theta) &= \int_E |\phi(x)|^2 [\Sigma^{-1} (x - \theta) (x - \theta)^* \Sigma^{-1} - \Sigma^{-1}] \exp\{-\frac{1}{2} (x - \theta)^* \Sigma^{-1} (x - \theta)\} dx \\
&= \int_E |\phi(x)|^2 \Sigma^{-1} (x - \theta) (x - \theta)^* \Sigma^{-1} \exp\{-\frac{1}{2} (x - \theta)^* \Sigma^{-1} (x - \theta)\} dx - \Sigma^{-1} v_{\text{int}}(\theta) ,
\end{aligned}$$

respectively. Therefore

$$(\log v_{\text{int}})'(\theta) = \frac{v'_{\text{int}}(\theta)}{v_{\text{int}}(\theta)} = \int_E (x - \theta)^* \Sigma^{-1} p(\theta, x) dx ,$$

and

$$\begin{aligned} (\log v_{\text{int}})''(\theta) &= \frac{v''_{\text{int}}(\theta)}{v_{\text{int}}(\theta)} - \left(\frac{v'_{\text{int}}(\theta)}{v_{\text{int}}(\theta)} \right)^* \frac{v'_{\text{int}}(\theta)}{v_{\text{int}}(\theta)} \\ &= \int_E \Sigma^{-1} (x - \theta) (x - \theta)^* \Sigma^{-1} p(\theta, x) dx \\ &\quad - \left[\int_E \Sigma^{-1} (x - \theta) p(\theta, x) dx \right] \left[\int_E (x - \theta)^* \Sigma^{-1} p(\theta, x) dx \right] - \Sigma^{-1} , \end{aligned}$$

with the probability density

$$p(\theta, x) \propto |\phi(x)|^2 \exp\left\{-\frac{1}{2} (x - \theta)^* \Sigma^{-1} (x - \theta)\right\} ,$$

parametrized by θ and known up to the normalizing constant $v_{\text{int}}(\theta)$, i.e.

$$p(\theta, x) = \frac{|\phi(x)|^2 \exp\left\{-\frac{1}{2} (x - \theta)^* \Sigma^{-1} (x - \theta)\right\}}{\int_E |\phi(x')|^2 \exp\left\{-\frac{1}{2} (x' - \theta)^* \Sigma^{-1} (x' - \theta)\right\} dx'} .$$

It follows from the Cauchy–Schwartz inequality that

$$(\log v_{\text{int}})''(\theta) \geq -\Sigma^{-1} \quad \text{hence} \quad (\log v)''(\theta) = \Sigma^{-1} + \frac{1}{2} (\log v_{\text{int}})''(\theta) \geq \frac{1}{2} \Sigma^{-1} \geq c I ,$$

in the sense of symmetric matrices (here, the positive constant c is half the smallest eigenvalue of the matrix Σ^{-1} , i.e. half the inverse of the largest eigenvalue $\sigma_{\max}$ of the covariance matrix Σ). The Hessian matrix $(\log v)''(\theta)$ is bounded from below by a definite positive matrix that does not depend on θ , hence the mapping $\theta \mapsto v(\theta)$ is strongly log-convex, and this property holds uniformly no matter what the function ϕ appearing in the definition of $v(\theta)$ may be. Notice also that

$$(\log v)'(\theta) = \theta^* \Sigma^{-1} + \frac{1}{2} (\log v_{\text{int}})'(\theta) = \frac{1}{2} \int_E (x + \theta)^* \Sigma^{-1} p(\theta, x) dx . \quad \square$$

Lemma A.3 *If the function ϕ is bounded, then the mappings $v_{\text{int}}(\theta)$ and $v_{\text{int}}^{1/2}(\theta)$ are bounded and globally Lipschitz continuous.*

PROOF. Clearly

$$0 \leq v_{\text{int}}(\theta) = \int_E |\phi(x)|^2 \exp\left\{-\frac{1}{2} (x - \theta)^* \Sigma^{-1} (x - \theta)\right\} \frac{dx}{\sqrt{\det(2\pi\Sigma)}} \leq \sup_{x \in E} |\phi(x)|^2 .$$

Recall that, keeping track of the normalizing constant

$$v'_{\text{int}}(\theta) = \int_E |\phi(x)|^2 (x - \theta)^* \Sigma^{-1} \exp\left\{-\frac{1}{2} (x - \theta)^* \Sigma^{-1} (x - \theta)\right\} \frac{dx}{\sqrt{\det(2\pi\Sigma)}} ,$$

hence

$$\begin{aligned} |v'_{\text{int}}(\theta)| &\leq \int_E |\phi(x)|^2 |(x-\theta)^* \Sigma^{-1}| \exp\{-\tfrac{1}{2}(x-\theta)^* \Sigma^{-1}(x-\theta)\} \frac{dx}{\sqrt{\det(2\pi\Sigma)}} \\ &\leq \sup_{x \in E} |\phi(x)|^2 \int_E |u^* \Sigma^{-1}| \exp\{-\tfrac{1}{2}u^* \Sigma^{-1}u\} \frac{du}{\sqrt{\det(2\pi\Sigma)}} , \end{aligned}$$

which shows the first statement. Using the Cauchy–Schwartz inequality yields

$$\begin{aligned} |v'_{\text{int}}(\theta)| &\leq \int_E |\phi(x)|^2 |(x-\theta)^* \Sigma^{-1}| \exp\{-\tfrac{1}{2}(x-\theta)^* \Sigma^{-1}(x-\theta)\} \frac{dx}{\sqrt{\det(2\pi\Sigma)}} \\ &\leq \left\{ \int_E |\phi(x)|^2 \exp\{-\tfrac{1}{2}(x-\theta)^* \Sigma^{-1}(x-\theta)\} \frac{dx}{\sqrt{\det(2\pi\Sigma)}} \right\}^{1/2} \\ &\quad \left\{ \int_E |\phi(x)|^2 |(x-\theta)^* \Sigma^{-1}|^2 \exp\{-\tfrac{1}{2}(x-\theta)^* \Sigma^{-1}(x-\theta)\} \frac{dx}{\sqrt{\det(2\pi\Sigma)}} \right\}^{1/2} \\ &\leq v_{\text{int}}^{1/2}(\theta) \sup_{x \in E} |\phi(x)| \left\{ \int_E |u^* \Sigma^{-1}|^2 \exp\{-\tfrac{1}{2}u^* \Sigma^{-1}u\} \frac{du}{\sqrt{\det(2\pi\Sigma)}} \right\}^{1/2} , \end{aligned}$$

hence

$$|(v_{\text{int}}^{1/2})'(\theta)| = \tfrac{1}{2} \frac{|v'_{\text{int}}(\theta)|}{v_{\text{int}}^{1/2}(\theta)} \leq \tfrac{1}{2} \sup_{x \in E} |\phi(x)| \left\{ \int_E |u^* \Sigma^{-1}|^2 \exp\{-\tfrac{1}{2}u^* \Sigma^{-1}u\} \frac{du}{\sqrt{\det(2\pi\Sigma)}} \right\}^{1/2} ,$$

which shows the second statement. $\square$

Remark A.4 Clearly, the mapping $\theta \mapsto \exp\{\tfrac{1}{2}\theta^* \Sigma^{-1}\theta\}$ is locally bounded and Lipschitz continuous, and it follows from Lemma A.3 that the mapping $v(\theta) = \exp\{\tfrac{1}{2}\theta^* \Sigma^{-1}\theta\} v_{\text{int}}^{1/2}(\theta)$ is locally Lipschitz continuous.

In many cases, the state variable x has two components, say x_o and x_{no} , and the function $\phi(x) = \phi(x_o)$ depends on the component x_o only. In such a case, the minimization problem can be further simplified, i.e. its dimension can be reduced.

Proposition A.5 *Let*

$$x = \begin{pmatrix} x_o \\ x_{\text{no}} \end{pmatrix} , \quad \theta = \begin{pmatrix} \theta_o \\ \theta_{\text{no}} \end{pmatrix} \quad \text{and} \quad \Sigma = \begin{pmatrix} \Sigma_o & \Sigma_{o,\text{no}} \\ \Sigma_{\text{no},o} & \Sigma_{\text{no}} \end{pmatrix} .$$

If the function $\phi(x) = \phi(x_o)$ depends on the component x_o only, then the unique minimizer of

$$v(\theta) = \exp\{\tfrac{1}{2}\theta^* \Sigma^{-1}\theta\} \left\{ \int_E |\phi(x)|^2 \exp\{-\tfrac{1}{2}(x-\theta)^* \Sigma^{-1}(x-\theta)\} dx \right\}^{1/2} ,$$

takes the form $\theta^{\bullet\bullet} = (\theta_o^{\bullet\bullet}, \theta_{\text{no}}^{\bullet\bullet})$ where $\theta_o^{\bullet\bullet}$ denotes the unique minimizer of

$$v_o(\theta_o) = \exp\{\tfrac{1}{2}\theta_o^* \Sigma_o^{-1}\theta_o\} \left\{ \int_{E_o} |\phi(x_o)|^2 \exp\{-\tfrac{1}{2}(x_o - \theta_o)^* \Sigma_o^{-1}(x_o - \theta_o)\} dx_o \right\}^{1/2} ,$$

and where $\theta_{\text{no}}^{\bullet\bullet} = \Sigma_{\text{no},o} \Sigma_o^{-1} \theta_o^{\bullet\bullet}$.

PROOF. Clearly

$$\begin{aligned} & \int_E |\phi(x)|^2 \exp\{-\tfrac{1}{2}(x-\theta)^* \Sigma^{-1}(x-\theta)\} dx \\ &= \int_{E_o} |\phi(x_o)|^2 \exp\{-\tfrac{1}{2}(x_o-\theta_o)^* \Sigma_o^{-1}(x_o-\theta_o)\} dx_o , \end{aligned}$$

up to a multiplicative constant that does not depend on θ , or keeping track of normalizing constants

$$\begin{aligned} & \int_E |\phi(x)|^2 \exp\{-\tfrac{1}{2}(x-\theta)^* \Sigma^{-1}(x-\theta)\} \frac{dx}{\sqrt{\det(2\pi\Sigma)}} \\ &= \int_{E_o} |\phi(x_o)|^2 \exp\{-\tfrac{1}{2}(x_o-\theta_o)^* \Sigma_o^{-1}(x_o-\theta_o)\} \frac{dx_o}{\sqrt{\det(2\pi\Sigma_o)}} . \end{aligned}$$

Introducing the Schur complement $\Delta = \Sigma_{no} - \Sigma_{no,o} \Sigma_o^{-1} \Sigma_{o,no}$ of the matrix Σ_o in the block-matrix Σ , it holds

$$\Sigma = \begin{pmatrix} \Sigma_o & \Sigma_{o,no} \\ \Sigma_{no,o} & \Sigma_{no} \end{pmatrix} = \begin{pmatrix} I & 0 \\ \Sigma_{no,o} \Sigma_o^{-1} & I \end{pmatrix} \begin{pmatrix} \Sigma_o & 0 \\ 0 & \Delta \end{pmatrix} \begin{pmatrix} I & \Sigma_o^{-1} \Sigma_{o,no} \\ 0 & I \end{pmatrix} ,$$

hence

$$\Sigma^{-1} = \begin{pmatrix} I & -\Sigma_o^{-1} \Sigma_{o,no} \\ 0 & I \end{pmatrix} \begin{pmatrix} \Sigma_o^{-1} & 0 \\ 0 & \Delta^{-1} \end{pmatrix} \begin{pmatrix} I & 0 \\ -\Sigma_{no,o} \Sigma_o^{-1} & I \end{pmatrix} ,$$

and

$$\begin{aligned} \theta^* \Sigma^{-1} \theta &= (\theta_{no} - \Sigma_{no,o} \Sigma_o^{-1} \theta_o)^* (\Sigma_{no} - \Sigma_{no,o} \Sigma_o^{-1} \Sigma_{o,no})^{-1} (\theta_{no} - \Sigma_{no,o} \Sigma_o^{-1} \theta_o) \\ &\quad + \theta_o^* \Sigma_o^{-1} \theta_o . \end{aligned}$$

Therefore

$$\begin{aligned} v(\theta) &= \exp\{\tfrac{1}{2} \theta^* \Sigma^{-1} \theta\} \left\{ \int_E |\phi(x)|^2 \exp\{-\tfrac{1}{2}(x-\theta)^* \Sigma^{-1}(x-\theta)\} dx \right\}^{1/2} \\ &= \exp\{\tfrac{1}{2} (\theta_{no} - \Sigma_{no,o} \Sigma_o^{-1} \theta_o)^* (\Sigma_{no} - \Sigma_{no,o} \Sigma_o^{-1} \Sigma_{o,no})^{-1} (\theta_{no} - \Sigma_{no,o} \Sigma_o^{-1} \theta_o)\} \\ &\quad \exp\{\tfrac{1}{2} \theta_o^* \Sigma_o^{-1} \theta_o\} \left\{ \int_{E_o} |\phi(x_o)|^2 \exp\{-\tfrac{1}{2}(x_o-\theta_o)^* \Sigma_o^{-1}(x_o-\theta_o)\} dx_o \right\}^{1/2} \\ &= \exp\{\tfrac{1}{2} (\theta_{no} - \Sigma_{no,o} \Sigma_o^{-1} \theta_o)^* (\Sigma_{no} - \Sigma_{no,o} \Sigma_o^{-1} \Sigma_{o,no})^{-1} (\theta_{no} - \Sigma_{no,o} \Sigma_o^{-1} \theta_o)\} v_o(\theta_o) , \end{aligned}$$

where

$$v_o(\theta_o) = \exp\{\tfrac{1}{2} \theta_o^* \Sigma_o^{-1} \theta_o\} \left\{ \int_{E_o} |\phi(x_o)|^2 \exp\{-\tfrac{1}{2}(x_o-\theta_o)^* \Sigma_o^{-1}(x_o-\theta_o)\} dx_o \right\}^{1/2} ,$$

up to a multiplicative constant that does not depend on θ . Clearly, the unique minimizer of $v(\theta)$ takes the form $\theta^{\bullet\bullet} = (\theta_o^{\bullet\bullet}, \theta_{no}^{\bullet\bullet})$ where $\theta_o^{\bullet\bullet}$ denotes the unique minimizer of $v_o(\theta_o)$ and where $\theta_{no}^{\bullet\bullet} = \Sigma_{no,o} \Sigma_o^{-1} \theta_o^{\bullet\bullet}$. $\square$

Finding the unique minimizer of the strongly convex mapping $\theta \mapsto \log v(\theta)$ is routinely transformed into finding the unique zero of the mapping $\theta \mapsto (\log v)'(\theta)$, and this can be achieved using a stochastic approximation algorithm. Indeed, recall that

$$(\log v)'(\theta) = \frac{1}{2} \int_E (x + \theta)^* \Sigma^{-1} p(\theta, x) dx ,$$

hence the (column vector) gradient has the following integral expression

$$\nabla(\log v)(\theta) = \frac{1}{2} \Sigma^{-1} \int_E (x + \theta) p(\theta, x) dx ,$$

with the probability density

$$p(\theta, x) \propto |\phi(x)|^2 \exp\{-\frac{1}{2} (x - \theta)^* \Sigma^{-1} (x - \theta)\} ,$$

parametrized by θ and known up to the (usually untractable) normalizing constant

$$v_{\text{int}}(\theta) = \int_E |\phi(x)|^2 \exp\{-\frac{1}{2} (x - \theta)^* \Sigma^{-1} (x - \theta)\} dx .$$

An iteration of the stochastic gradient algorithm would take the form

$$\theta_{n+1} = \theta_n - \gamma_n H_n ,$$

where H_n is some random possibly biased approximation of the gradient $\nabla(\log v)(\theta_n)$. This class of stochastic approximation algorithms is studied in Appendix B.

B Stochastic approximation

To summarize the situation, here are some features of the minimization problem presented at the end of Appendix A

- the objective function $\ell(\theta)$ is strongly convex (hence it has a unique minimizer θ_{opt}), but it does not have an integral expression,
- its gradient does have an integral expression of the form

$$\nabla \ell(\theta) = \int_E H(\theta, x) p(\theta, x) dx ,$$

- the probability density $p(\theta, x) = \pi(\theta, x)/Z(\theta)$ depends on the parameter θ , and it is known up to some untractable normalizing constant

$$Z(\theta) = \int_E \pi(\theta, x) dx .$$

Indeed, the connection with the minimization problem presented at the end of Appendix A is made through the following identification

$$\begin{aligned}\ell(\theta) &= \log v(\theta) , & H(x, \theta) &= \frac{1}{2} \Sigma^{-1} (x + \theta) , \\ \pi(\theta, x) &= |\phi(x)|^2 \exp\{-\frac{1}{2} (x - \theta)^* \Sigma^{-1} (x - \theta)\} & \text{and} & \quad Z(\theta) = v_{\text{int}}(\theta) .\end{aligned}$$

The key step in a stochastic gradient algorithm for this minimization problem is to provide at each iteration of the algorithm an approximation of the gradient $\nabla \ell(\theta_n)$ by some random variable H_n , for instance

- A if an *oracle* is available, or if it is possible to sample directly from the current probability density $p(\theta_n, x)$, e.g. using an accept/reject method, then set

$$H_n = \frac{1}{N_n} \sum_{j=1}^{N_n} H(\theta_n, X_j) ,$$

an unbiased estimator of $\nabla \ell(\theta_n)$, where $(X_1, \dots, X_{N_n})$ are i.i.d. random variables with common probability density $p(\theta_n, x)$,

- B if it is not possible to sample directly from the current probability density $p(\theta_n, x)$, but it is possible to sample from a dominating probability density $\rho(\theta_n, x)$, then introduce the ratio

$$w(\theta_n, x) = \frac{\pi(\theta_n, x)}{\rho(\theta_n, x)} \quad \text{for any } x \in E,$$

and set

$$H_n = \frac{\sum_{j=1}^{N_n} H(\theta_n, X_j) w(\theta_n, X_j)}{\sum_{j=1}^{N_n} w(\theta_n, X_j)} \quad \text{and} \quad Z_n = \frac{1}{N_n} \sum_{j=1}^{N_n} w(\theta_n, X_j) ,$$

a biased estimator of $\nabla \ell(\theta_n)$ and an unbiased estimator of $Z(\theta_n)$ respectively, where $(X_1, \dots, X_{N_n})$ are i.i.d. random variables with common probability density $\rho(\theta_n, x)$,

- C alternatively, if it is not possible to sample directly from the current probability density $p(\theta_n, x)$, but it is possible to sample from a probability transition density $M_n(x, x')$ that admits the current probability density $p(\theta_n, x)$ as its unique invariant density, e.g. using a Metropolis–Hastings algorithm, then set

$$H_n = \frac{1}{N_n} \sum_{j=1}^{N_n} H(\theta_n, X_j) ,$$

a biased estimator of $\nabla \ell(\theta_n)$, where $(X_1, \dots, X_{N_n})$ is a Markov chain with some initial probability density and with probability transition density $M_n(x, x')$.

Nemirovskii et al. [2] considers the case A, Atchadé et al. [1] considers the case C, while the focus here is on the case B. The estimator is updated as

$$\theta_{n+1} = \theta_n - \gamma_n H_n ,$$

hence the design parameters are the gain sequence $\{\gamma_n\}$ and the Monte Carlo batch-size sequence $\{N_n\}$. Write

$$H_n = \nabla \ell(\theta_n) + \varepsilon_n ,$$

where ε_n denotes the approximation error, and let

$$B_n = \mathbb{E}[\varepsilon_n \mid \mathcal{F}_n] \quad \text{and} \quad M_n = \mathbb{E}[|\varepsilon_n|^2 \mid \mathcal{F}_n] ,$$

denote the conditional bias and the conditional mean-square error, respectively. If importance sampling (case B) is used, then both the bias B_n and the mean-square error M_n are of order $O(1/N_n)$.

Bound on the gradient approximation Clearly

$$|H_n|^2 = |\nabla \ell(\theta_n)|^2 + 2 \varepsilon_n^* \nabla \ell(\theta_n) + |\varepsilon_n|^2 .$$

If the approximation error is unbiased, i.e. if $\mathbb{E}[\varepsilon_n \mid \mathcal{F}_n] = 0$, then

$$\mathbb{E}[|H_n|^2 \mid \mathcal{F}_n] = |\nabla \ell(\theta_n)|^2 + M_n ,$$

hence

$$\mathbb{E}|H_n|^2 = \mathbb{E}|\nabla \ell(\theta_n)|^2 + \mathbb{E}[M_n] .$$

Otherwise, if the approximation error is biased, then using the Young inequality yields

$$\begin{aligned} \mathbb{E}[|H_n|^2 \mid \mathcal{F}_n] &= |\nabla \ell(\theta_n)|^2 + 2 B_n^* \nabla \ell(\theta_n) + M_n \\ &\leq (1 + A) |\nabla \ell(\theta_n)|^2 + \frac{|B_n|^2}{A} + M_n , \end{aligned}$$

hence

$$\mathbb{E}|H_n|^2 \leq (1 + A) \mathbb{E}|\nabla \ell(\theta_n)|^2 + \frac{\mathbb{E}|B_n|^2}{A} + \mathbb{E}[M_n] .$$

and taking $A = a/N_n$ yields

$$\mathbb{E}|H_n|^2 \leq (1 + \frac{a}{N_n}) \mathbb{E}|\nabla \ell(\theta_n)|^2 + N_n \frac{\mathbb{E}|B_n|^2}{a} + \mathbb{E}[M_n] .$$

Approximation of the minimizer Clearly

$$\theta_{n+1} - \theta_{\text{opt}} = \theta_n - \theta_{\text{opt}} - \gamma_n H_n \quad \text{with} \quad H_n = \nabla \ell(\theta_n) + \varepsilon_n ,$$

and strong convexity, see Nesterov [3, Theorem 2.1.9], yields

$$(\theta_n - \theta_{\text{opt}})^* \nabla \ell(\theta_n) = (\theta_n - \theta_{\text{opt}})^* (\nabla \ell(\theta_n) - \nabla \ell(\theta_{\text{opt}})) \geq c |\theta_n - \theta_{\text{opt}}|^2 ,$$

hence

$$\begin{aligned}
|\theta_{n+1} - \theta_{\text{opt}}|^2 &= |\theta_n - \theta_{\text{opt}}|^2 - 2\gamma_n (\theta_n - \theta_{\text{opt}})^* \nabla \ell(\theta_n) \\
&\quad - 2\gamma_n (\theta_n - \theta_{\text{opt}})^* \varepsilon_n + \gamma_n^2 |H_n|^2 \\
&\leq (1 - 2\gamma_n c) |\theta_n - \theta_{\text{opt}}|^2 - 2\gamma_n (\theta_n - \theta_{\text{opt}})^* \varepsilon_n + \gamma_n^2 |H_n|^2 .
\end{aligned}$$

If the approximation error is unbiased, i.e. if $\mathbb{E}[\varepsilon_n | \mathcal{F}_n] = 0$, then

$$\mathbb{E}[|\theta_{n+1} - \theta_{\text{opt}}|^2 | \mathcal{F}_n] \leq (1 - 2\gamma_n c) |\theta_n - \theta_{\text{opt}}|^2 + \gamma_n^2 \mathbb{E}[|H_n|^2 | \mathcal{F}_n] ,$$

hence

$$\mathbb{E}|\theta_{n+1} - \theta_{\text{opt}}|^2 \leq (1 - 2\gamma_n c) \mathbb{E}|\theta_n - \theta_{\text{opt}}|^2 + \gamma_n^2 \mathbb{E}|H_n|^2 .$$

This is equation (2.8) in Nemirovskii et al. [2]. Otherwise, if the approximation error is biased, then using the Young inequality yields

$$\begin{aligned}
\mathbb{E}[|\theta_{n+1} - \theta_{\text{opt}}|^2 | \mathcal{F}_n] &\leq (1 - 2c\gamma_n) |\theta_n - \theta_{\text{opt}}|^2 \\
&\quad - 2\gamma_n (\theta_n - \theta_{\text{opt}})^* B_n + \gamma_n^2 \mathbb{E}[|H_n|^2 | \mathcal{F}_n] \\
&\leq (1 - (2c - A)\gamma_n) |\theta_n - \theta_{\text{opt}}|^2 \\
&\quad + \gamma_n \frac{|B_n|^2}{A} + \gamma_n^2 \mathbb{E}[|H_n|^2 | \mathcal{F}_n] ,
\end{aligned}$$

and taking $A = c$ yields

$$\mathbb{E}|\theta_{n+1} - \theta_{\text{opt}}|^2 \leq (1 - \gamma_n c) \mathbb{E}|\theta_n - \theta_{\text{opt}}|^2 + \gamma_n \frac{\mathbb{E}|B_n|^2}{c} + \gamma_n^2 \mathbb{E}|H_n|^2 .$$

Approximation of the minimum value It would be possible, following Nemirovskii et al. [2] or Atchadé et al. [1], to get bounds for the approximation of the minimum value of the mapping $\ell(\theta) = \log v(\theta)$, i.e. to bound the approximation error $\ell(\theta_n) - \ell(\theta_{\text{opt}})$. However, it would still remain to explain how to compute $\ell(\theta_n)$ in practice, and besides, recall that the ultimate objective here is to get bounds for the approximation of the minimum value of the mapping $v(\theta) = \exp\{\theta^* \Sigma^{-1} \theta\} v_{\text{int}}^{1/2}(\theta)$. Clearly

$$\exp\{\tfrac{1}{2} \theta_n^* \Sigma^{-1} \theta_n\} Z_n^{1/2} - v(\theta_{\text{opt}}) = \exp\{\tfrac{1}{2} \theta_n^* \Sigma^{-1} \theta_n\} (Z_n^{1/2} - v_{\text{int}}^{1/2}(\theta_n)) + v(\theta_n) - v(\theta_{\text{opt}}) ,$$

and using the bound $(\sqrt{z} - \sqrt{v})^2 \leq |z - v|$ yields

$$\begin{aligned}
|\exp\{\tfrac{1}{2} \theta_n^* \Sigma^{-1} \theta_n\} Z_n^{1/2} - v(\theta_{\text{opt}})|^2 &\leq 2 \exp\{\theta_n^* \Sigma^{-1} \theta_n\} |Z_n^{1/2} - v_{\text{int}}^{1/2}(\theta_n)|^2 + 2 |v(\theta_n) - v(\theta_{\text{opt}})|^2 \\
&\leq 2 K^2 |Z_n - v_{\text{int}}(\theta_n)| + 2 L^2 |\theta_n - \theta_{\text{opt}}|^2 ,
\end{aligned}$$

hence

$$\mathbb{E}|\exp\{\tfrac{1}{2} \theta_n^* \Sigma^{-1} \theta_n\} Z_n^{1/2} - v(\theta_{\text{opt}})|^2 \leq 2 K^2 \{ \mathbb{E}|Z_n - v_{\text{int}}(\theta_n)|^2 \}^{1/2} + L^2 \mathbb{E}|\theta_n - \theta_{\text{opt}}|^2 .$$

Titre : Poursuite multi-cibles sur signal brut

Mots clés : poursuite multi-cibles, radar, filtrage particulaire, crossover, conception optimale

Résumé : Le sujet de la thèse est la poursuite de plusieurs cibles, qui évoluent de manière indépendante, à partir de mesures recueillies par des capteurs ponctuels ou sous la forme d'une image-radar. Les algorithmes étudiés entrent dans la catégorie des filtres particulaires.

Dans une première partie, on compare deux algorithmes proposés dans la littérature pour la poursuite multi-cibles. L'algorithme ATRAPP proposé par Ubada-Medina *et al.* permet en particulier de générer des particules multi-cibles brassées (un vecteur multi-cibles dans lequel chaque cible est possiblement répliqué à partir d'une particule multi-cibles différente), au prix d'une hypothèse d'indépendance a posteriori des cibles, presque jamais vérifiée dans les situations pratiques.

Dans une deuxième partie, on introduit un algorithme original, utilisant une transition markovienne auxiliaire et généralisant le principe du filtre particulaire auxiliaire. Dans le contexte de la poursuite multi-cibles, ce nouvel algorithme permet de générer des particules multi-cibles brassées, sans hypothèse d'indépendance a posteriori. Le choix optimal des paramètres de conception (poids auxiliaire, matrice ou densité de transition auxiliaire) est discuté. L'utilisation d'une densité de transition auxiliaire permet de diversifier plus largement les particules multi-cibles et de diminuer encore la variance. Le choix optimal se réduit à la résolution numérique de problèmes d'optimisation continus. Dans le cas particulier, important en pratique, où la dynamique des cibles est linéaire gaussienne, on montre que ces problèmes sont fortement convexes.

Title: Bayesian multitarget tracking from raw data

Keywords: multitarget tracking, radar, particle filtering, crossover, optimal design

Abstract: This thesis is about tracking several independent targets, using measurements collected by sensors or from a radar-image. The algorithms studied belong to the class of particle filters.

In a first part, two algorithms proposed in the literature for multitarget tracking are compared. The ATRAPP algorithm proposed by Ubada-Medina *et al.* makes it possible to generate shuffled multitarget particles (a multitarget vector where each target is possibly replicated from a different multitarget particle), under a posterior target independence assumption that is almost never met in practical situations.

In a second part, an original algorithm is introduced, that uses an auxiliary Markov transition in its design and generalizes the auxiliary particle filter principle. In the context of multitarget tracking, this new algorithm makes it possible to generate shuffled multitarget particles, with no posterior target independence assumption. The optimal choice of the design parameters (auxiliary weights, auxiliary transition matrix or kernel) is discussed. Using an auxiliary transition kernel makes it possible to better shake the multitarget particles and to further decrease the variance. The optimal design reduces to the numerical solution of continuous optimization problems. In the special case where targets have a linear Gaussian dynamics, a special case of practical interest in applications, it is shown that these problems are strongly convex.