Multiple-Object Estimation Techniques for Challenging Scenarios

A dissertation presented by

Flávio Eler De Melo

to The Department of Electrical Engineering and Electronics, and Computer Science

in partial fulfillment of the requirements
for the degree of

Doctor of Philosophy

in the subject of

Computer Science

University of Liverpool
Liverpool, United Kingdom
November 2017

Thesis advisor: Professor Simon R. Maskell Flávio Eler De Melo

Multiple-Object Estimation Techniques for Challenging Scenarios

Abstract

A series of methods for solving the multi-object estimation problem in the context sequential Bayesian inference is presented. These methods concentrate on dealing with challenging scenarios of multiple target tracking, involving fundamental problems of nonlinearity and non-Gaussianity of processes, high state dimensionality, high number of targets, statistical dependence between target states, and degenerate cases of low signal-to-noise ratio, high uncertainty, lowly observable states or uninformative observations. These difficulties pose obstacles to most practical multi-object inference problems, lying at the heart of the shortcomings reported for state-of-the-art methods, and so elicit novel treatments to enable tackling a broader class of real problems.

The novel algorithms offered as solutions in this dissertation address such challenges by acting on the root causes of the associated problems. Often this involves essential dilemmas commonly manifested in Statistics and Decision Theory, such as trading off estimation accuracy with algorithm complexity, soft versus hard decision, generality versus tractability, conciseness versus interpretativeness etc. All proposed algorithms constitute stochastic filters, each of which is formulated to address specific aspects of the challenges at hand, while offering tools to achieve judicious compromises in the aforementioned dilemmas.

Two of the filters address the weight degeneracy observed in sequential Monte Carlo filters, particularly for nonlinear processes. One of these filters is designed for nonlinear non-Gaussian high-dimensional problems, delivering representativeness of the uncertainty in high-dimensional states while mitigating part of the inaccuracies that arise from the curse of dimensionality. This filter is shown to cope well with scenarios of multimodality, high state uncertainty, uninformative observations and high number of false alarms. A multi-object filter deals with the problem of considering dependencies between target states in a way that is scalable to a large number of targets, by resorting to probabilistic graphical structures. Another multi-object filter treats the problem of reducing computational complexity of a state-of-the-art cardinalized filter to deal with large number of targets, without compromising accuracy significantly. Finally, a framework for associating measurements across observation sessions for scenarios of low state observability is proposed, with application to an important Space Surveillance task: cataloging of space debris in the geosynchronous/geostationary belt.

The devised methods treat the considered challenges by bringing about rather general questions, and provide not only principled solutions, but also analyzes the essence of the investigated problems, extrapolating the implemented techniques to a wider spectrum of similar problems in Signal Processing.

I dedicate this work to the "Maker of heaven and earth, the sea, and everything in them," in the name of the Lord Jesus, for His glory and honor.

Acknowledgement

Having conducted and documented this work was both a great learning experience and a dignifying endeavor. Firstly, I thank God for my life, my health, for His provision, enabling and for the blessing of this opportunity. It is fair to say that this work could not be accomplished without valuable contributions from a number of people. I would like to thank the University of Liverpool, both as an institution and in the figure of its staff, who offered me this great opportunity and provided me with the means necessary to carry out this work. Also, I would like to thank my thesis advisor, Simon Maskell, who contributed not only as a great educator, proposing research topics, clarifying many of my doubts and showing different perspectives, but also as a very supportive friend in difficult times. I am also indebted to a number of colleagues for their help, with ideas, suggestions, criticism and encouragement. Those with direct contributions are Matteo Fasiolo, Chinmaya Mishra, Chongyang Liu, Yifan Zhou, Frederick Daum, Emmanuel Delande, Nicholas Harwood, but many others contributed indirectly in some way. Finally, I would like to thank my wife, Pauline, for her support, patience and care, which were nothing less than essential.

Contents

List of Figures

Abbreviations

AIS Annealed Importance Sampling.

AR Admissible Region.

BFGS Broyden-Fletcher-Goldfarb-Shanno algorithm.

CAR Constrained Admissible Region.

CAR-MHF Constrained Admissible Region, Multiple Hypothesis Filter.

CDF Cumulative distribution function.

CPDA Coupled Probabilistic Data Association.

CPHD Cardinalized Probability Hypothesis Density.

CPU Central Processing Unit.

DG-CPHD Discrete-Gamma Cardinalized Probability Hypothesis Density.

DISP Distinguishable and Independent Stochastic Populations.

EHM Efficient Hypothesis Management.

EKF Extended Kalman Filter.

ESS Effective Sample Size.

FDA Functional Data Analysis.

FISST Finite Set Statistics.

GEO Geosynchronous/Geostationary Earth Orbit.

GH Gauss-Hermite.

GHD Generalized Hyperbolic Distribution.

GHKF Gauss-Hermite Kalman Filter.

GHPF Gauss-Hermite Particle Filter.

GHQ Gauss-Hermite Quadrature.

GLMB Generalized Labeled Multi-Bernoulli.

GNN Global Nearest Neighbor Data Association.

GPF Gaussian Particle Flow.

GPGPU General-Purpose Graphics Processing Units.

HMC Hamiltonian Monte Carlo.

HMM Hidden Markov model.

IDM Intelligent Driver Model.

IMF-EKF Information Matrix Fusion, Extended Kalman Filter.

IOD Initial Orbit Determination.

IS Importance Sampling.

JIPDA Joint Integrated Probabilistic Data Association.

JIPDA-EHM Joint Integrated Probabilistic Data Association with Efficient Hypothesis Management.

JIPDA-GNN Joint Integrated Probabilistic Data Association with Global Nearest Neighbor Data Association.

JPDA Joint Probabilistic Data Association.

JPDA-EHM Joint Probabilistic Data Association with Efficient Hypothesis Management.

JPDA-EHM3 Joint Probabilistic Data Association with Efficient Hypothesis Management 3.

JPDA-GM Joint Probabilistic Data Association, Gaussian mixture.

JPDA-GM-EHM Joint Probabilistic Data Association, Gaussian Mixture with Efficient Hypothesis Management.

JPDAC Joint Probabilistic Data Association Coupled.

JSD Jensen-Shannon Divergence.

KL Kullback-Leibler.

LEO Low Earth Orbit.

LMB Labeled Multi-Bernoulli.

LMCO Langevin Monte Carlo with Ozaki discretization.

MALA Metropolis-adjusted Langevin algorithm.

MAP Maximum a posteriori.

MAPF Marginal Auxiliary Particle Filter.

MBPF Marginal Bootstrap Particle Filter.

MCMC Markov chain Monte Carlo.

MeMBer Multi-target Multi-Bernoulli.

MEPF Marginal EKF-based Particle Filter.

MH-JPDA Multiple-Hypothesis Joint Probabilistic Data Association.

MHF Multiple Hypothesis Filter.

MHT Multiple Hypothesis Tracking.

ML Maximum likelihood.

MMSE Minimum mean square error.

MMSE-MAP Minimum mean square error (state), maximum a posteriori (association).

MMSE-ML Minimum mean square error (state), maximum likelihood (association).

MOSPA Mean Optimal Subpattern Assignment.

MPF Marginal Particle Filter.

MSE Mean square error.

MTT Multiple Target Tracking.

MUPF Marginal UKF-based Particle Filter.

NEES Normalized estimation error squared.

OD Orbit Determination.

ODE Ordinary Differential Equation.

OSPA Optimal Subpattern Assignment.

PDA Probabilistic Data Association.

PDE Partial Differential Equation.

PDF Probability density function.

PF Particle Filter.

PGF Probability generating function.

PGFL Probability generating functional.

PHD Probability Hypothesis Density.

RFS Random Finite Set.

RMSE Root mean square error.

ROC Receiver Operating Characteristics.

RSO Resident Space Object.

SDE Stochastic Differential Equation.

SDPF Scaled-drift Particle Flow.

SIR Sequential Importance Resampling.

SIS Sequential Importance Sampling.

SMC Sequential Monte Carlo.

SmHMC Sequential manifold Hamiltonian Monte Carlo.

SmMALA Sequential manifold Metropolis-Adjusted Algorithm.

SPF Stochastic Particle Flow.

SPF-GS Stochastic Particle Flow, Gaussian sum.

SPF-MPF Stochastic Particle Flow, Marginal Particle Filter.

SSA Space Situational Awareness.

UCT Uncorrelated Track.

UKF Unscented Kalman Filter.

UPF Unscented Particle Filter.

Nomenclature

α	Acceptance probability (Ch. 1 and 2); index of vehicles in IDM (Ch. 2); location parameter (Ch. 2); shape parameter of the Gamma distribution (Ch. 4); right ascension (Ch. 5 and App. A)
$\bar{\mathrm{x}}_{S,k}$, $\mathrm{P}_{S,k}$	Sample mean and empirical covariance matrix
$\bar{n}_{\max}$	Maximum number of terms for approximating derivatives of the predicted discrete-Gamma probability generating functional
$\bar{w}_a$	Multidimensional quadrature weight with index a
$\bar{w}_j$	Gauss-Hermite quadrature weight with index j
β	Parameter of the spatial correlation matrix (Ch. 2); rate parameter of the Gamma distribution (Ch. 4)
$\boldsymbol{\beta}$	Fitted coefficients to a parametric curve (orbital arc)
$\mathbf{q}$	Position vector of the observation site
$\mathbf{r}$	Geocentric position vector
$\boldsymbol{\rho}$	Line of sight vector
$\breve{\mathrm{x}}^{(a)}$	Transformed multidimensional grid abscissa with index a
$\breve{w}_k^{(i)}$	Unnormalized importance weight of sample i at time step k
δ	Infinitesimal variation; declination (Ch. 5 and App. A)
δ_{x}	Dirac-delta at x
δ_{ij}	Kronecker delta
$\eta_c(\cdot)$, $\mathrm{c}(\cdot)$	Spatial distribution of false alarms
γ, $\gamma(\cdot)$	Parameter of the Generalized Hyperbolic distribution (Ch. 2); Euler-Mascheroni constant (Ch. 4); parametric curve (Ch. 5)
$\Gamma_u(\cdot)$	Incomplete Gamma function
$\hat{\pi}(\cdot)$, $\tilde{\pi}(\cdot)$	Approximated target distribution
$\hat{N}_k$	Estimated number of targets at time step k
$\hat{q}(\cdot)$, $\tilde{q}(\cdot)$	Approximated proposal distribution
λ	Pseudo-time (Ch. 2); rate of false alarms (Ch. 4)
λ_{fa}	Rate of false alarms
λ_c, $\lambda_{V,\mathrm{fa}}$	Rate of clutter density
$\mathbb{J}_i(\cdot)$	Bessel function of first kind, of order i
$\mathbb{K}_i[\cdot]$	Modified Bessel function of second kind, of order i
$\imath$	Orbit inclination
$\boldsymbol{\imath}$	Argument of latitude
$\mathcal{M}^{\mathfrak{a}}[N^{(i)}]$	Set of measurements directly assigned to node $N^{(i)}$
$\mathcal{D}$	Matrix of information gain due to representing pairs of targets as dependent

$\mathcal{E}$	Set of edges of a graph
$\mathcal{F}(\mathcal{X})$	σ-algebra on the state vector space
$\mathcal{F}(\mathcal{Y})$	σ-algebra on the observation vector space
$\mathcal{H}_{\mathrm{x}}[\cdot]$	Hessian matrix with respect to x
$\mathcal{I}_n$	Index set containing n distinct natural numbers
$\mathcal{J}_{\mathrm{x}}[\cdot]$, $\mathrm{J}_{\mathrm{x}}[\cdot]$	Jacobian matrix with respect to x
$\mathcal{L}[\cdot]$	Forward Kolmogorov operator
$\mathcal{L}[\lambda]$, $\mathcal{L}_\lambda$	Langevin diffusion at pseudo-time instant λ
$\mathcal{L}\{\cdot\}(s)$	Laplace transform
$\mathcal{M}_i$	Set of validated measurements for the ith target
$\mathcal{P}$	Probability measure
$\mathcal{P}_\pi$	Target probability measure
$\mathcal{R}$	Matrix of (binary) relations between pairs of targets
$\mathcal{S}$	Region of the state space
$\mathcal{S}_j$	Physical location of the jth sensor
$\mathcal{T}(\cdot)$	Deterministic filtering map
$\mathcal{V}$	Set of vertices (nodes) of a graph
$\mathcal{X}$	State vector space
$\mathcal{Y}$	Observation vector space
$\mathfrak{a}(\cdot)$	Measurement-to-node association map (Ch. 3); measurement-to-measurement association map (Ch. 5)
$\mathfrak{h}(\cdot)$	Markov chain speed function
B	Input matrix (Ch. 2 and App. A); matrix for computing bounds of fitted coefficients (Ch. 5)
D	Diffusion matrix
F, A	State transition matrix
G	Rao tensor metric
H	Observation matrix
K	Kalman gain matrix
Q	State process noise covariance matrix
R	Observation noise covariance matrix
S	Innovation covariance matrix
u_k, ξ_k	State process noise vector at discrete time step k
v_k, υ_k	Observation noise vector at discrete time step k
w_t, W_t	Standard Wiener process vector at time instant t
$\mathrm{x}(t)$, x_t	State vector at time instant t
$\mathrm{x}^{(a)}$	Multidimensional grid abscissa with index a
x_k	State vector at discrete time step k
$\mathrm{x}_k^{(i)}$	Sample (particle) i at time step k
$\mathrm{y}(t)$, y_t	Observation vector at time instant t
Y_k	Set of observations at time step k
y_k	Observation vector at discrete time step k
$\mathscr{V}_{k,i}$	Validation region for the ith target
μ	Mean, drift vector (Ch. 2); Earth gravitational constant (Ch. 5 and App. A)

$\mu_m^{(i)}$, $\Sigma_m^{(i)}$	Mean vector and covariance matrix of the ith mixture component
μ_N, σ_N^2	Cardinality mean and variance
ν	True anomaly (Ch. 5)
Ω	Sample space (Ch. 1 and 2); right ascension of the ascending node (Ch. 5 and App. A)
ω	Argument of perigee
$\overline{\mathrm{de}}(i)$	Set of nodes that are not i or any of its descendents
ϕ	Joint association event (Ch. 2)
$\pi(\cdot)$	Target distribution
Ψ	Random finite set of observations
$\psi(\cdot)$	Graph factor
$\sigma_{m,i}(\cdot)$	Elementary homogeneous symmetric function of degree i over m elements
σ_m	Permutation of m elements
σ_q^2, σ_r^2	State process and observation noise variance parameters
$\mathrm{an}(i)$	Set of ancestors of node i
$\mathrm{ch}(i)$	Set of children of node i
$\mathrm{cs}(i)$	Set of cousins of node i, composed of all non-descendents nodes that are not ancestors of i
$\mathrm{de}(i)$	Set of descendents of node i
$\mathrm{pa}(i)$	Set of parents of node i
$\mathrm{sb}(i)$	Set of siblings of node i: $\mathrm{sb}(i) \triangleq \mathrm{ch}(\mathrm{pa}(i)) \setminus \{i\}$
Θ	Polynomial terms for computing the posterior intensity function (DG-CPHD); state vector of orbital elements (Ch. 5)
θ	Elevation (Ch. 1); association event (Ch. 2); polynomial terms for computing cardinality moments (Ch. 4)
$\tilde{\mathcal{L}}[\lambda]$, $\tilde{\mathcal{L}}_\lambda$	Approximated Langevin diffusion at pseudo-time instant λ
$\tilde{p}^{(i)}(\cdot)$	Locally approximated prior density based on the ith mixture component
ε	Desired accuracy (Ch. 2); probability that the cardinality is within a specified interval (Ch. 4)
$\varphi(\cdot)$	Test function, function to be estimated
$\varsigma(\cdot)$	Single-target spatial probability density
Ξ	Random finite set of target states
$\{\mathcal{F}_k\}_{k\in\mathbb{N}_0}$	Filtration of the σ-algebra $\mathcal{F}$ over discrete time steps
$\{\mathcal{F}_t\}_{t\geq 0}$	Filtration of the σ-algebra $\mathcal{F}$ over continuous time
a	Index of abscissae (Ch. 1); acceleration (Ch. 2); measurement-to-target association map (Ch. 3); semi-major axis (Ch. 5 and App. A)
a_k	Association random variable (process) that maps targets to measurements, at time step k
c_k	Mixture index random variable that indicates the component of a target's state at time step k
D_k	Detection event at time step k (Ch. 3); intensity function (Ch. 4)
d_k	Detection random variable at time step k
d_x	Dimension of vector x
e	Eccentricity (first eccentricity)

E_k	Existence event at time step k
e_k	Existence random variable at time step k
$f(\cdot)$, $\mathrm{f}(\cdot)$	State function; test function of a probability generating functional (Ch. 4)
G	Graph (Ch. 3)
$g(\cdot)$, $\mathrm{g}(\cdot)$	Test function; function of the state process noise (Ch. 3))
$G(\cdot)$, $G[\cdot]$	Probability generating function; probability generating functional (Ch. 4)
$h(\cdot)$, $\mathrm{h}(\cdot)$	Observation function; test function of a probability generating functional (Ch. 4)
H_j, $\breve{H}_j$	Hermite polynomial of order j
k	Discrete time step
L	Number of integration steps (Ch. 2); factors for approximating derivatives of polylogarithms (Ch. 4)
M	Number of grid nodes per dimension (Ch. 1); number of measurements (Ch. 3 and 4); mean anomaly (Ch. 5)
N	Number of samples (Ch. 1 and 2); number of targets (Ch. 3 and 4)
n	Cardinality random variable (Ch. 4); mean motion (Ch. 5)
$N^{(i)}$	Node in a graph corresponding to the ith target
$n_{\max}$	Maximum number of terms for the cardinality probability mass function (CPHD)
N_b	Number of newborn targets
N_d	Number of detections
N_m	Number of measurements
N_s	Number of sensors (Ch. 2); number of survived targets (Ch. 4)
N_t	Number of targets
p	Position; semi-latus rectum (Ch. 5)
$p(\cdot)$	Probability density function
$p_\ell(\mathrm{y}\|\cdot)$, $\ell_{\mathrm{y}}(\cdot)$	Likelihood function for observation y
$p_{\mathrm{birth}}(\cdot)$	Spatial distribution of a newborn target
$p_\Xi(\cdot)$	Multi-target probability density
p_b	Probability of birth
$p_b(\cdot)$	Probability distribution of birth cardinality (number of newborn targets)
$p_c(\cdot)$	Probability distribution of clutter cardinality (number of false alarms)
P_d, p_d	Probability of detection
$p_N(\cdot)$	Probability distribution of cardinality (target number)
P_o	Probability that an object is observed more than once
p_s	Probability of survival
$p_t(\cdot\|\cdot)$, $P_t(\cdot,\cdot)$	Markov transition kernel
$p_x(\cdot)$	Prior probability density function
$q(\cdot)$	Proposal distribution
q_d	Probability of misdetection
S	Probability current
T	Time horizon (Ch. 2); simulation horizon (Ch. 3), fixed sampling time $\Delta t \coloneqq T$
t	Continuous time instant
V	Surveillance region volume
v	Velocity
$w_{\pi,k}^{(i)}$	Mixture weight of the ith target component at time step k

w_B	Backward weights
w_F	Forward weights
$w_k^{(i)}$	Normalized importance weight of sample i at time step k
x_j	One-dimensional Gauss-Hermite abscissa with index j

0

Introduction

Multiple target tracking concerns estimating the states of various objects in a scene. This is performed based on a hypothesis of how each object moves (state model) and how the sensor observes the states (sensor model), by incorporating information from collections of noisy observations, under the possibility of misdetections and false alarms (a.k.a. clutter), and uncertainty about the origin of each measurement. The standard formulation of the problem is probabilistic, referring to the states and observations as random processes and the state estimates are taken in some optimal sense[1], usually resorting to the Bayesian statistics formalism.

Although multi-target tracking (MTT) is an established field and innumerous methods exist, the task remains far from easy in practical terms. In usual applications, the typical scenario (and technology) imposes the observation of targets' states as unable to provide a mechanism to identify the origin of each measurement in its association to a real detected object. Thus, besides the Bayesian inference to be carried out, the practitioner is left with a complex decision of which candidate measurements should be incorporated into the state estimation of each object. This gives rise to a data association problem, since, in order to jointly estimate the states of all targets, under the uncertainty of which measurement pertains to which object, various measurement-to-target association hypotheses are possible. Distinct approaches to this problem lead to different classes of estimators:

I. A standard class of filters where the decision is explicitly posed as a combinatorial problem (e.g., Probabilistic Data Association [6]).

II. A class of filters where the decision is avoided by estimating the state of all objects jointly in a coupled way, by considering all measurements to update the joint multi-target state (e.g., Coupled Probabilistic Data Association [16]).

III. A class of filters where the description of the multi-object state and observation processes is given as *random finite sets* [131] or *stochastic populations* [143], internalizing permutations of objects to implicitly account for the universe of measurement-to-target associations (e.g., Probability Density Hypothesis [126]).

Indeed, the scenario where several targets are reported in the same neighborhood, with the possibility of clutter, is complicated. In the literature, this condition is commonly understood as resulting in a *persistent interference*, caused by neighboring targets, along with a *random interference*, caused by clutter among the measurements [7]. When targets are close to each other, there is a strong interdependence of the measurement-to-target associations, as illustrated by Figure 1. In the illustration, three targets currently in custody are predicted to be in the positions of T1, T2 and T3, respectively. At the current time step, a sensor collects five

[1] Suboptimal in some approximations.

measurements at the positions of M1, M2, M3, M4 and M5. The ellipses in Figure 1 represent the uncertainties (covariance matrices) of T1, T2 and T3, enclosing regions for a certain level of confidence. In this scene, the following measurement origins could be possible:

H1. Measurement M1 was generated from target T2 or is clutter;

H2. Measurement M2 was generated from target T2, or target T1, or is clutter;

H3. Measurement M3 was generated from target T1 or is clutter;

H4. Measurement M4 was generated from target T1, or target T3, or is clutter;

H5. Measurement M5 was generated from target T3 or is clutter.

The interdependence becomes clear when one observes that, for instance, if M2 originated from target T1, then it is likely that M1 originated from target T2. The usual assumption that produces the set of hypotheses H1–H5 is that each measurement may have originated from at most one target or is a false alarm. Extended hypotheses can also be formulated under more general premises, by either considering that multiple targets may generate only one measurement (merged measurement model), as a result of targets falling in the same resolution cell of the sensor, or that a single target may generate a collection of measurements (extended target model) when it occupies more than one resolution cell. Sensors with finite resolution are subject to these possibilities and expand the scope of the standard measurement model.

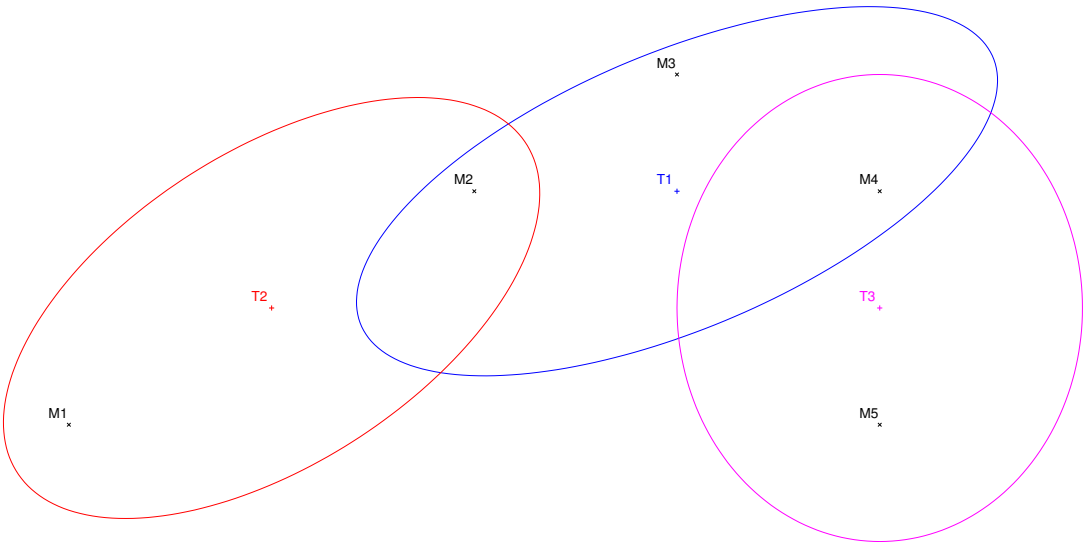


Figure 1: Example of several targets in the same neighbourhood

In general, multiple target trackers propose estimates in the minimum mean square error (MMSE) sense [7]. Filters of class I advocate an approach by which the conditional expectation of the state is taken considering marginal probabilities (or likelihoods) of events in a set of mutually and exhaustive association hypotheses. Some variants in class I propose the MMSE state estimate with maximum a posteriori association (MMSE-MAP), which selects the association event with highest posterior probability as a realization, on which the expectation of the state is conditioned. A similar approach proposes MMSE state estimate with maximum

likelihood association (MMSE-ML). Heuristic approaches are also common, such as the *nearest neighbor* that selects which measurement is associated with which target based on a distance metric. Filters of class II are built purely in the MMSE sense, without the need for an explicit solution of the data association problem. Filters of class III also avoid the data association problem but require a special treatment for producing estimates. They are based on a function that consolidates the whole multi-target state configuration, the intensity function[2] [126], and so filters of class III require extraction of peaks or distinguishable components of this function.

0.1 Multi-object filtering problem

Let us formalize the multi-object filtering problem. Consider that the state of each individual object evolves with time, denoted as $t \in \mathbb{R}_+$, according to a stochastic process, $\{\mathrm{x}_t\}_{t\geq 0}$, $\mathrm{x}_t \in \mathcal{X} \subseteq \mathbb{R}^{d_x}$, where d_x is the number of elements of x_t. Observables about each object are realized according to another stochastic process, $\{\mathrm{y}_t\}_{t\geq 0}$, $\mathrm{y}_t \in \mathcal{Y} \subseteq \mathbb{R}^{d_y}$, where d_y is the number of elements of y_t. Since the description is given in terms of random processes, it is usual to fix the underlying probability space, $(\Omega, \mathcal{F}, \mathcal{P})$, where Ω is the sample space, $\mathcal{F}$ is a σ-algebra on Ω that accounts for the universe of possible events, and $\mathcal{P}$ is a probability measure that assigns probabilities to events. In addition, the description considers the measurable spaces $(\mathcal{X}, \mathcal{F}_t(\mathcal{X}))$ and $(\mathcal{Y}, \mathcal{F}_t(\mathcal{Y}))$ of the state and observation processes, respectively, with σ-algebras $\mathcal{F}_t(\mathcal{X}) \triangleq \sigma\{\mathrm{x}_t | t \geq 0\}$ and $\mathcal{F}_t(\mathcal{Y}) \triangleq \sigma\{\mathrm{y}_t | t > 0\}$ that compose the filtrations $\{\mathcal{F}_t(\mathcal{X})\}_{t\geq 0}$ and $\{\mathcal{F}_t(\mathcal{Y})\}_{t\geq 0}$ of the σ-algebra $\mathcal{F}$. The filtering problem concerns characterizing the object states by estimates conditioned on observations. In the optimal filtering formalism, the problem consists in finding the best $\mathbb{L}^2$-estimate of some $\mathcal{F}_t(\mathcal{X})$-measurable function of the state process, $\varphi(\mathrm{x}_t)$, based on the observations, that is, one wishes to find an $\mathcal{F}_t(\mathcal{Y})$-measurable quantity $\hat{\varphi}_t$ that minimizes $\|\hat{\varphi}_t - \varphi(\mathrm{x}_t)\|_2$. The formal solution to this problem is generally given by $\hat{\varphi}_t = \mathbb{E}[\varphi(\mathrm{x}_t)|\mathcal{F}_t(\mathcal{Y})]$ admitting almost sure equivalence, where $\mathbb{E}$ denotes the expectation with respect to the probability measure $\mathcal{P}$. In practice, observations are acquired at discrete time instants and so can be thought of as random sequences $\{\mathrm{y}_k\}_{k\in\mathbb{N}}$ realized at $t = 1\Delta t, \ldots, k\Delta t$. Similarly, as the state process shall be characterized conditioned on the observations, the states can also be denoted as discrete-time random sequences $\{\mathrm{x}_k\}_{k\in\mathbb{N}_0}$.

Now consider a number n of targets in a multiple target context. By indexing all targets in the scene by an index set $\mathcal{I}_{n,k} \coloneqq \{i_1, \ldots, i_n\} \subset \mathbb{N}$, each target state is described as $\mathrm{x}_k^{(i)}$ such that $i \in \mathcal{I}_{n,k}$. The set of all targets in the scene at time step k can be denoted as $\mathrm{x}_k^{(1:n)} = \mathrm{x}_k^{\mathcal{I}_{n,k}} \coloneqq \{\mathrm{x}_k^{(i)} | i \in \mathcal{I}_{n,k}\}$. Similarly, a number m of observations of targets' states are collected, indexed by $\mathcal{I}_{m,k} \coloneqq \{j_1, \ldots, j_m\} \subset \mathbb{N}$, and so each measurement can be identified as $\mathrm{y}_k^{(j)}$ such that $j \in \mathcal{I}_{m,k}$. The complete set of measurements is denoted as $\mathrm{Y}_k \triangleq \mathrm{y}_k^{(1:m)} \coloneqq \{\mathrm{y}_k^{(j)} | j \in \mathcal{I}_{m,k}\}$ and we write the history of observations up to time step k as $\mathrm{Y}_{1:k} \coloneqq \{\mathrm{Y}_1, \ldots, \mathrm{Y}_k\}$. Multi-object state and observation spaces can be denoted as the Cartesian products $\mathcal{X}^n = \prod_{i\in\mathcal{I}_{n,k}} \mathcal{X}$ and $\bar{\mathcal{Y}}^m = \prod_{j\in\mathcal{I}_{m,k}} \bar{\mathcal{Y}}$ respectively, where $\bar{\mathcal{Y}} \triangleq \mathcal{Y} \cup \mathcal{Y}_{\mathrm{fa}}$ and, in this context, the observation process is assumed to admit false alarms $\mathrm{y}_{k,\mathrm{fa}}^{(j)} \in \mathcal{Y}_{\mathrm{fa}}$ as realizations and missed detections may be present ($m \leq n$).

In the multi-object framework, the ultimate goal of any multiple-target tracker is to infer estimates about target states conditioned on information provided by the observation history,

[2] First order moment of the associated point process.

that is, obtaining

$$\mathcal{P}_k[\varphi] = \mathbb{E}\left[\varphi(\mathrm{x}_k^{(1:n)})|\mathcal{F}_k(\bar{\mathcal{Y}}^m)\right] \equiv \int_{\mathcal{X}^n} \varphi(\mathrm{x}_k^{(1:n)})\mathcal{P}_k(d\mathrm{x}_k^{(1:n)}, \mathrm{Y}_{1:k}), \tag{1}$$

for a test function $\varphi : \mathcal{X}^n \to \mathcal{X}'$, measurable in $\mathcal{F}_k(\mathcal{X}^n) = \sigma\{\mathrm{x}_t^{(1:n)}|0 < t \leq k\Delta t\}$, where $\mathcal{P}_k(\cdot, \mathrm{Y}_{1:k})$ is a posterior multi-target probability measure, and $\mathrm{x}_k^{(1:n)}$ should be understood in terms of the joint event $\mathrm{X}_k^{(1:n)} = \bigcap_{i=1}^n \mathrm{X}_k^{(i)}$ in the measurable space $(\mathcal{X}^n, \mathcal{F}_k(\mathcal{X}^n))$. In practical terms, computing estimates as in (1) involves expressing the posterior multi-target measure explicitly, which constitutes the main scope of most multiple target trackers.

0.2 Some challenges in multiple target tracking

In the standard setting, the multi-target tracking problem manifests itself as hybrid, characterized by a continuous uncertainty in the multi-target state space, and by a discrete uncertainty arising from the measurement-to-target association ambiguities. A challenge arises when one tries to track a large number of targets in this setting because a high computational effort is demanded to solve the association ambiguities. This is particularly true when the problem is set in a combinatorial form, where an exhaustive enumeration of data association hypotheses must be performed. Usual solutions, as per class I, involve techniques to manage the number of hypotheses [163] or mixing the marginal data association components to consolidate the state estimates [73]. An alternative approach, per class III, deals with this challenge by interpreting the multi-target state process as a composite random process: either in the form of a *random finite set* [131], a *point process* [44], or a *stochastic population* [143]. In this interpretation, realizations of these processes are symmetric (unordered) and the filtering procedure internalizes all association possibilities such that the data association has not to be explicitly treated. The drawbacks of this alternative approach are either loss of the ability to keep target identities [126], increased computational complexity when incorporating labels in the multi-target state process [164], or increased computational complexity when maintaining distinguishable hypotheses [58].

A typical assumption of the multi-target tracking problem considers that target states are mutually independent conditioned on the observations. In general, this assumption simplifies the multi-target state estimation, since in that case the uncertainty of each target state can be fully represented independently of the others, and the joint multi-target state probability measure is the product of the individual target measures. In the standard class I this assumption plays an essential role as the target states are represented and updated separately, and in class III it is fundamental to enable a formulation that requires *points*[3] of the joint multi-target state process to be mutually independent. Filters in class II relax the independence assumption by estimating the target states jointly along with their dependencies, but often at the expense of higher complexity and nonscalability to large numbers of targets. The presumed target independence is reasonable in a number of cases, when targets are well separated for most of the time, but it may result in relevant inaccuracies when targets get into proximity and stay in this condition for a considerable duration of time, largely due to the interdependences illustrated in Figure 1. Representing mutual target dependencies accurately while maintaining

[3] In the sense of point processes.

scalability in terms of numbers of targets constitutes an important challenge for the field.

From a different perspective, an aspect that poses another challenge to the multi-target tracking problem is the effect of nonlinearity and non-Gaussianity in the sequential evolution of the multi-target state uncertainty. Nonlinearity is present in the mathematical description of a wide range of physical phenomena, usually appearing as a model is devised for high representation fidelity or for explaining second and higher-order effects. In contrast, non-Gaussianity is related to the nature of the uncertainty source, and may appear due to many factors including, not exhaustively, spatial constraints (e.g., a robot moving close to a wall), dynamical constraints (e.g., an object orbiting the Earth), nonlinear combination of two or more uncertainty sources (e.g., uncertainty in range due to noise in Cartesian components), inherent asymmetry (e.g., counting numbers of objects must consider only nonnegative integers), etc.

Representing the probability distributions that delineate the target state uncertainty can be achieved in closed form for only a few special cases (e.g., linear, Gaussian state and observation processes). In the general case, representing the associated probability distributions requires a comprehensive description to some accuracy level, which must also be concise enough to enable computational tractability. One could think in terms of analytical requirements to represent smooth probability density functions, and note that an exact representation would only be possible in terms of function series with infinitely many terms[4]. Hence, the challenge lies in approximating the state uncertainty by a finite number of parameters, under a structured model, which must be judiciously chosen. This is usually achieved by means of approximations based on the Kalman filter [108] (e.g., Extended Kalman filter), mixtures (e.g., [185]), grid methods (e.g., [21]) or sequential Monte Carlo methods (e.g., [83]). Each of these methods reacts differently to two fundamental (statistical) principles: (i) dimensionality, whose increase dilutes representativity of the uncertainty by a finite number of parameters (*curse of dimensionality*), and (ii) inhomogeneity, by which the uncertainty poses distinct demands of representativity for different regions of the state space *(heteroscedasticity*). As a consequence, state-of-the-art filters have their particular shortcomings, and hence show characteristics with problem-specific suitability while opportunities for better uncertainty representations and filtering techniques remain open.

In cases marked by observations with low signal-to-noise ratio, both the challenges of solving data association ambiguity and accurate uncertainty representation are exacerbated. This happens because low signal-to-noise ratio involves high measurement noise variances, low probability of detection or a large number of false alarms. In fully probabilistic frameworks this scenario renders hypotheses where each measurement might originate from a target or a false alarm with comparable probabilities. In some extreme cases, the probability of a measurement being a false alarm can be much higher than that of originating from a real target (e.g., multistatic sonar applications). In these cases, the resulting posterior multi-target probability measure and the corresponding intensity function are very diffuse, consisting of several modes that are significant for the description, which calls forth the need for an accurate uncertainty representation. As an illustration, think of the relative difficulty of approximating a multimodal measure in comparison to approximating a single mode. Concomitantly, the uncertainty about the origin of each measurement is very high, either because the observation likelihood is very uninformative (due to large noise variances or small probability of detection), or because

[4]This is the criterion to tell if the function is analytic.

of a relatively large number of false alarms that leads to a relevant number of equally likely hypotheses. For filters whose complexity is dominated by the number of measurements (e.g., [127]), the case with a high number of false alarms is particularly difficult, and these filters may become infeasible for scenarios with a moderate to large number of targets.

A similar problem appears in situations where the state observability is low. It is important to note that observability is a characteristic mostly related to the observation settings (sensor coordinate frame, observation period etc) rather than to the object's attributes or to the observation noise. In those situations, measurements may be well accurate and yet offer insufficient information for the state estimation to be accomplished. This is true to a great extent in Space Surveillance applications, where short-periodic observation of geostationary satellites (and debris) by electro-optical instruments provide accurate angular measurements but offers little information about which orbit the objects describe. Specifically for this example, the settings impose that, in order to determine the orbit of the observed objects, more than one observation batch must be obtained in different sessions separated by hours to days. Since multiple objects appear in the observations, the scenario elicits a technique for associating measurements from different observation sessions, i.e., solving the measurement-to-measurement association problem. This problem is very important as its solution facilitates cataloging space debris in the context of Space Surveillance for supporting new satellite launching and maintaining safe operations. The problem is as challenging as important since the association ambiguities are enormous, hundreds of objects are observed in short-term campaigns, and purely heuristic strategies (e.g., nearest neighbor) are prone to discard most of the redundant information contained in multiple observation sessions.

0.3 Thesis Statement and Contributions

This dissertation contributes to some important subdomains of Computer Science and Signal Processing, namely, multi-target tracking and data fusion, stochastic filtering and sequential Monte Carlo methods, dynamical modeling, and to some specific methods of Celestial Mechanics for orbit determination. The dissertation proposes techniques at the boundary between target tracking with sensor data fusion and modern pre-eminent techniques in stochastic filtering and probabilistic transport, and also explores the interconnection of multiple target tracking with probabilistic graphical inference to contribute to the enhancement of the status quo of filtering with data fusion, with applications including, but not limited to, maritime, terrestrial, aerial and space surveillance. As a primary objective, this dissertation works on and tests the following hypothesis.

Hypothesis. *State-of-the-art multi-object estimators are frequently studied under benign scenarios, driven by well-characterized problems, under mild conditions, such that reported successes of established algorithms are largely dependent on the problem settings. In the literature, it is often the case that these estimators are demonstrated to perform satisfactorily for well-known problems and standard settings, possibly outperforming some other methods, however, the very important cases and settings for which the estimators don't apply or will not perform appropriately are virtually not explored, and commonly unknown. This typical circumstance leads to algorithm evaluations that predominantly demonstrate adequacy to the stated problem (and its specificities) while revealing only a small subset of inherent characteristics and involved*

principles in the filtering techniques.

We propose that by exploring extreme cases, where the fundamental shortcomings of estimators are exposed via exacerbation of the problem scenario, mathematical principles and root causes of defects can be analyzed to a greater extent, which not only clarifies important questions about the methods but also establishes the context for novel methods to be proposed. We also propose that, with such a deeper understanding at hand, adopting methodologies from other scientific domains and elaborating on mathematical principles to work around essential imperfections may successfully solve many estimation challenges. Ultimately, in this context, we propose a series of novel methods, each of which addresses a subset of aspects that limit the performance of state-of-the-art estimators, and extends the applicability to relevant challenging scenarios.

It is worth noting that the correctness or falsehood of this hypothesis may not admit a formal proof. Nevertheless, this dissertation is focused on providing evidence, in an empirical sense, for or against its validity. It does so by analyzing limitations of state-of-the-art filters under challenging multi-target tracking scenarios and introducing a series of methods to deal with those challenges, treating problems of nonlinearity and non-Gaussianity of processes, high state dimensionality and large numbers of targets, statistical dependence between target states, low signal-to-noise ratio, and lowly observable states.

Each chapter presents a novel estimation technique to deal with one or more challenging scenarios described in the previous section, and so each contribution of this dissertation is organized in a single chapter. Each chapter is self-contained, providing a detailed overview of the problem context, literature review, problem statement, mathematical derivations, numerical experiments, discussion, conclusions and the bibliographical references used within the chapter. In this structure, the reader may consider each chapter as a completely independent article, and no specific order is necessary to understand the content. Appendix A is an account of a well-established method for discretization of linear state-space models from the author's perspective. Chapter 6 summarizes and concludes the dissertation. The contributions of the dissertation are briefly described as follows.

Chapter 1: Hybrid Gauss-Hermite filter

The first contribution is a filter for treating the problem of representing the target state uncertainty for highly nonlinear processes while partially avoiding a limitation of filters based on sequential importance sampling (particle filter). In Chapter 1 this novel algorithm for sequential Bayesian estimation is presented, consisting of a hybrid method that combines a particle-based representation of the prior state uncertainty with an efficient grid-based method to estimate the posterior probability density. The proposed filter uses a prior Monte Carlo empirical measure to induce a probability mass function that approximates the posterior probability distribution. Such an approximation enables accurate numerical integration, by means of the Gauss-Hermite quadrature, to compute the state estimates and covariance matrices.

Chapter 2: Stochastic Particle Flow

The second contribution involves an in-depth analysis of the problem of representing the multi-target state uncertainty for high-dimensional nonlinear problems, which is as difficult as important. As a result, the contribution proposes a sophisticated method that capitalizes on modern

findings on a variational formulation of uncertainty propagation and transport-based filters, to represent high-dimensional nonlinear processes accurately. In addition, in Chapter 2, the method addresses successfully the challenges of tracking dependent targets by estimating all targets states jointly and excels in scenarios with low signal-to-noise ratio.

This novel method is based on an alternative way of performing Bayesian updates, by defining a continuum of densities between the prior probability density and the posterior, i.e., the filtering density. The novel filter uses equally weighted samples, each of which is associated with a local solution of the Fökker-Planck equation. This hybrid of Monte Carlo and local parametric approximation gives rise to a global approximation of the filtering density of interest. In Chapter 2, we show that, when compared with state-of-the-art methods, the Gaussian-mixture implementation of the new filtering technique, which we call Stochastic Particle Flow, has utility in the context of benchmark nonlinear high-dimensional filtering problems.

Chapter 3: Multi-target tracking of dependent targets

The contribution of Chapter 3 provides a scalable multi-target tracking framework that considers dependence between target states due to association uncertainty, and enables track management. This method builds on a target tracker previously developed to efficiently maintain target dependence structures, incorporating mechanisms for initiating new tracks via birth process and by inferring existence probabilities to cater for track confirmation and deletion. The proposed framework does not maintain a joint probability distribution over all the target states but rather relies on target dependence trees on which discrete probability distributions of adjacent targets are calculated. These dependence trees constitute probabilistic graphical models, where each node represents a target-related discrete random variable and the edges represent the probabilistic relations between nodes. Each node variable can take different values, one for each mixture component of the corresponding target state, with probabilities given by a discrete marginal distribution, and each edge is described by a conditional probability distribution relating two nodes. These probabilistic trees are predicted and updated at each time step such that the information lost in the approximation is minimal. For computing the marginal association probabilities, an exact, very efficient algorithm known as Efficient Hypothesis Management (EHM) is adopted in its most general form, which enables computations for pairs of dependent targets.

Chapter 4: Discrete-Gamma Cardinalized Probability Hypothesis Density

The fourth contribution is in the context of efficiently filtering large numbers of targets, and catering for low signal-to-noise scenarios while also estimating the number of targets (cardinality) in the scene in a fully probabilistic way. The resulting filter is a computationally efficient approximation of the Cardinalized Probability Hypothesis Density (CPHD) filter, one of the most acclaimed algorithms for multi-target Bayesian filtering due to its ability to accurately estimate the number of objects and the object states. The CPHD filter generalizes the Probabilistic Hypothesis Density (PHD) filter by jointly propagating the first-order multi-target moment (intensity function) along with the entire probability distribution on the number of targets (cardinality distribution). In general, the CPHD recursion is computationally intractable, however, successful approximations have been devised with reported computational complexity dominated by $\mathcal{O}(m^3)$ operations per filtering iteration, where m is the number of measurements.

In Chapter 4, the idea of approximating the cardinality distribution by two-parameter distributions is explored to provide an efficient approximation of the CPHD where the cardinality distribution is modeled as a discretized Gamma distribution.

Chapter 5: Measurement-to-measurement association for Space Surveillance

The contribution described in Chapter 5 treats the problem of probabilistic inference under low state observability (in the orbital elements space). The problem is stated in the context of a Space Surveillance application and has been solved by an algorithm that performs measurement-to-measurement associations across multiple observation sessions. Angles-only orbit determination of nearly geosynchronous and geostationary objects based on short-arc observations is challenging both because of their low observability and measurement uncertainty commensurate with the observed arc length. In general, the scenario requires multiple observation sessions, several hours apart, to deliver enough observability and reduced susceptibility of orbit determination to measurement noise. Ultimately, this obliges that observations from different sessions be correctly associated under the presence of other objects. In this chapter we propose a new framework to effectively associate observations from multiple sessions for uncorrelated tracks. The proposed framework is based on a new initial orbit determination method that enables a reasonable description of nearly-geosynchronous and geostationary orbits and their uncertainties, and a procedure for statistical comparison between estimated orbits in a space comprising orbital elements and measurements. The comparison generates likelihood values that quantify the similarity of observations across sessions. The new initial orbit determination algorithm is based on Escobal's method along with estimates of orbital elements by a new unperturbed two-body angular motion model.

Appendix A: Discretization of linear state-space models

The contribution of Appendix A is literary. It gives a brief account, in the perspective of the author, of an established method to discretize linear state space models. The method is well documented in the Control Engineering literature, however, apart from few exceptions, it is rarely discussed in the multi-target tracking literature. Appendix A provides a simple derivation of the procedure to discretize a continuous-time linear state-space model. The intent is clarifying how discrete-time models widely used in multi-target and multi-sensor tracking can be obtained from continuous-time models, by resorting to transformations in the Laplace domain and other techniques for solving stochastic differential equations.

References

[6] Y. Bar-Shalom, F. Daum, and J. Huang. "The Probabilistic Data Association Filter". In: *IEEE Control Systems* 29.6 (Dec. 2009), pp. 82–100.

[7] Y. Bar-Shalom, P. K. Willett, and X. Tian. *Tracking and Data Fusion: A Handbook of Algorithms*. YBS Publishing, Apr. 2011.

[16] H. A. P. Blom and E. A. Bloem. "Probabilistic data association avoiding track coalescence". In: *IEEE Transactions on Automatic Control* 45.2 (Feb. 2000), pp. 247–259.

[21] R. S. Bucy and K. D. Senne. "Digital Synthesis of Non-linear Filters". In: *Automatica* 7.3 (May 1971), pp. 287–298.

[44] D. J. Daley and D. Vere-Jones. *An Introduction to the Theory of Point Processes*. Vol. Volume I: Elementary Theory and Methods. Probability and its Applications. New York, Berlin, Paris: Springer, 2003.

[58] E. Delande, J. Houssineau, and D. E. Clark. "Multi-object filtering with stochastic populations". In: *ArXiv e-prints* (Jan. 2015). arXiv: 1501.04671 [stat.AP].

[73] T. E. Fortmann, Y. Bar-Shalom, and M. Scheffe. "Multi-target tracking using joint probabilistic data association". In: *19th IEEE Conference on Decision and Control including the Symposium on Adaptive Processes*. Dec. 1980, pp. 807–812.

[83] N. J. Gordon, D. J. Salmond, and A. F. M. Smith. "Novel approach to nonlinear/non-Gaussian Bayesian state estimation". In: *IEEE Proceedings - Radar and Signal Processing* 140.2 (Apr. 1993), pp. 107–113.

[108] R. E. Kalman. "A New Approach to Linear Filtering and Prediction Problems". In: *Transactions of the ASME–Journal of Basic Engineering* 82.D (1960), pp. 35–45.

[126] R. P. S. Mahler. "Multitarget Bayes Filtering via First-Order Multitarget Moments". In: *IEEE Transactions on Aerospace and Electronic Systems* 39.4 (Oct. 2003), pp. 1152–1178.

[127] R. P. S. Mahler. "A theory of PHD filters of higher order in target number". In: *Proc. SPIE 6235, Signal Processing, Sensor Fusion, and Target Recognition XV*. Vol. 6235. May 2006, 62350K–62350K-12.

[131] R. P. S. Mahler. *Statistical Multisource-Multitarget Information Fusion*. Norwood, MA, USA: Artech House, Inc., 2007.

[143] J. E. Moyal. "The General Theory of Stochastic Population Processes". In: *Acta Mathematica* 108 (1962), pp. 1–31.

[163] D. B. Reid. "An Algorithm for Tracking Multiple Targets". In: *IEEE Transactions on Automatic Control* 24.6 (Dec. 1979), pp. 843–854.

[164] S. Reuter et al. "The Labeled Multi-Bernoulli Filter". In: *IEEE Transactions on Signal Processing* 62.12 (June 2014), pp. 3246–3260.

[185] H. W. Sorenson and D. L. Alspach. "Recursive Bayesian Estimation Using Gaussian Sums". In: *Automatica* 7.4 (July 1971), pp. 465–479.

1

Hybrid Gauss-Hermite Filter

This chapter is intended to address one of the challenges identified in Chapter 0, namely, that of representing the probability distributions for highly nonlinear and non-Gaussian processes in an effective way while avoiding well-known defects in sequential Monte Carlo representations. To that end, we present an algorithm for sequential Bayesian estimation consisting of a hybrid method that combines a particle-based representation of the prior state uncertainty with an efficient grid-based method to estimate the posterior probability density. The proposed filter uses a prior Monte Carlo empirical measure to induce a probability mass function that approximates the posterior probability distribution. Such an approximation enables accurate numerical integration, by means of the Gauss-Hermite quadrature, to compute the state estimates and covariance matrices. The filter is prone to estimation errors dominated by the same approximation errors as those found in conventional particle filters, but it is well suited to generally solve nonlinear non-Gaussian filtering problems without the well-known *weight degeneracy* problem[1]. Simulation results demonstrate the versatility of the filter for practical problems, showing performance similar to that of particle filters with optimal proposal densities, for nonlinear non-Gaussian dynamic state-space models, with the advantage that the weight degeneracy problem is absent.

1.1 Overview

Probabilistic inference based on state-space models with stochastic transitions and partially observable states are well described by hidden Markov models (HMM). The generality of Markov models enables a description of stochastic processes in continuous state spaces that is applicable to a wide variety of problems in Statistics, Computer Science and Engineering. Nevertheless, these models lead to intractability in practice because, except for a few special cases, analytic solutions for exact inference are impossible for processes with continuous distributions. Particle filters arose as a class of sequential Monte Carlo algorithms for obtaining approximate solutions to inference problems, expressed in the framework of Markov models, for which analytic solutions cannot be achieved [83, 4, 66]. These algorithms are based on recursive Monte Carlo simulations of the state model at hand, via importance sampling, to provide weighted samples (also known as particles) that represent the probability density of the state process. In contrast to standard approximating methods for nonlinear systems, such as the Extended Kalman Filter (EKF), particle filters have the advantage that they do not rely on parametric statistics, local linearization for uncertainty propagation or crude functional approximations. The counterpart of this advantage is a higher computational demand when compared to filters similar to the

[1] As opposed to the *particle degeneracy* that is present in all sequential Monte Carlo filters due to the curse of dimensionality.

Kalman filter, and the statistical anomalies that arise from the sequential utilization of the importance sampling technique.

Two fundamental problems of particle filtering are manifested. The first is the susceptibility of these filters to the curse of dimensionality [46], sometimes referred to as "particle degeneracy": in favor of implementation simplicity, in their standard setting, particle filters propagate paths of samples in such a way that, at each step, they occupy a space with augmenting dimensionality [110], and so the finite set of samples can only represent the state process to an increasingly sparse extent. The second problem is the well-known "weight degeneracy" problem [66, 166]: a sequential decrease in accuracy of the uncertainty representation due to the inevitably increasing variance of the approximation. Typically, this problem is explained in the context of the mismatch between the probability distribution from which the samples are taken (proposal or importance distribution), and the actual distribution that the filter aims to represent (target distribution). The weight degeneracy originates from the fact that, for any finite number of proposed samples, only a fraction of them may populate the region of interest, that is, the region of high probability under the target distribution. Thus, the weights of samples far from the high-probability region not only become ineffective for producing estimates but also affect the global accuracy of the target description. The problem becomes exacerbated as sequential moves of the proposed samples further increase the inaccuracy of the uncertainty representation, which is reflected in a high variance of the importance weights and, eventually, all but one weight collapse to insignificant values.

Standard techniques devised to mitigate the degeneracy problem concentrate on constraining the weight's variance increase, consisting mainly of

i) resampling, which selects and reproduces the most effective samples,

ii) design of better-suited proposal distributions,

iii) use of information from the observation to determine which particles should survive resampling (predictive likelihood), e.g., Auxiliary Particle Filter [156], and

iv) methods for particle diversification, e.g., Markov chain Monte Carlo moves [66, 26] and regularization [145].

Furthermore, in addition to a multitude of existing variations of the sequential importance sampling algorithm, there are innumerous alternative techniques that have been introduced to address particle filtering shortcomings, either by transforming the proposal density [136], applying adaptive methods [34, 118], using accept/reject mechanisms [55, 26], incorporating intermediate proposals between filtering steps [146, 12], or transforming the particles through deterministic operations that avoid importance sampling completely, e.g., particle flow filters [49, 47] and transport-based filters [161].

In this chapter we present a filter that poses estimation of the posterior probability distribution as an explicit integration over points of a state-space grid, operating directly on a marginal space, which is akin to the marginal particle filter [110]. Moreover, the proposed filter proceeds estimation in a way similar to that of usual grid-based methods [11, 4, 178]. Nevertheless, the filtering method presented herein is conceptually hybrid as it makes use of a grid-based estimation technique set up by propagating Monte Carlo samples that express the previous posterior distribution. Essentially, the filtering principle is not underpinned by

importance sampling so that weight degeneracy is avoided. We depart from a Monte Carlo representation of the prior probability distribution to induce a posterior density discretized at specific points in the state space. These points are conveniently set at the roots of a Hermite polynomial, of an adequate order, to enable numerical integration over the posterior space by the Gauss-Hermite quadrature.

Sequential Bayesian estimation for nonlinear filtering relying on quadrature properties for numerical integration was explored by [21], which proposed an algorithm that stores a point-mass representation of the posterior density on a floating rectangular grid. A class of filters based on the Gauss-Hermite quadrature (GHQ) was introduced in [98] to approximate the posterior distribution by a single Gaussian density, capturing effects of higher-order moments by using selected points in a way similar to that of the Unscented Kalman Filter (UKF) [107] and its variants. In this class of filters, the GHQ is used to evaluate linear regression coefficients that parameterize a Gaussian to represent the posterior density, which explains its name coined as the "Gauss-Hermite (Kalman) filter" (GHKF). In addition, a Gaussian mixture was parameterized to extend the Gauss-Hermite Kalman filter to non-Gaussian processes [97, 98]. The original GHKF shares similarities with the UKF in the sense that they are based on computations over chosen state points (sigma-points) to propagate Gaussian densities through nonlinear filtering steps. This similarity has been explored in [210] by applying the GHKF to provide a suboptimal proposal density for a standard particle filter, giving rise to the Gauss-Hermite Particle Filter (GHPF).

A recursive nonlinear filter that relies on explicit numerical integration with respect to Gaussian densities has been proposed in [115]. This filter provides estimates for a diffusion process with discrete-time observations, and it depends on an iterative centering procedure of the integration grid for the quadrature to be performed. Another algorithm that performs GHQ explicitly to tackle estimation of multimodal densities is presented in [171]. This algorithm minimizes the Kullback-Leibler (KL) divergence to ensure matching between the true posterior density and a proposal density. With a different perspective, an important filter presented in [28] solves the Fökker-Planck equation using the Generalized Edgeworth Series (GES) expansion of the evolving densities, and evaluates the Bayes' rule using the GHQ. Another recent filter approximates multi-dimensional integrals to perform nonlinear Bayesian estimation [102]. This filter utilizes weighted sparse-grid quadrature points to approximate the integrals in the estimation algorithm, determining locations and weights of the univariate quadrature points by the moment matching method.

The nonlinear filter presented in this chapter is unique in the sense that it uses samples to propagate and construct the posterior density at the GHQ abscissae. The estimation problem is solved without the use of techniques for distribution matching, without specific procedures of grid centering or refinement, and without parameterization of densities via statistical linear regression or sigma-points. In addition, neither it assumes that the process and measurement noises are Gaussian, nor that densities are unimodal.

The chapter is organized into eight sections. Section 1.2 poses the stochastic filtering problem and introduces sequential Monte Carlo filters. In Section 1.3 we present the formulation of a probability mass function that approximates the posterior probability density function (pdf) of a Markov process. The posterior probability density is approximated at points of interest in the state space, given a previous posterior density described by a set of samples that represent

a Monte Carlo empirical measure. In Section 1.5 we describe the Gauss-Hermite quadrature procedure and explain an established method for obtaining Hermite polynomial roots for any region of the state space. Section 1.4 derives the sampling method used to close the filtering recursion. Sections 1.6 and 1.7 assemble the building blocks to construct the hybrid Gauss-Hermite filter and outline the algorithm. In Section 1.8 we provide two simulation examples and performance results, and Section 1.9 concludes the chapter.

1.2 Stochastic filtering

1.2.1 Filtering problem

Consider a system whose (hidden) states evolve with time, denoted as $t \in \mathbb{R}_+$, according to a stochastic process, $\{\mathrm{x}_t\}_{t\geq 0}$, $\mathrm{x}_t \in \mathcal{X} \subseteq \mathbb{R}^{d_x}$, where d_x is the number of elements of x_t. Observables about the system are realized according to another stochastic process, $\{\mathrm{y}_t\}_{t\geq 0}$, $\mathrm{y}_t \in \mathcal{Y} \subseteq \mathbb{R}^{d_y}$, where d_y is the number of elements of y_t. Since the description is given in terms of random processes, it is usual to fix the underlying probability space, $(\Omega, \mathcal{F}, \mathcal{P})$, where Ω is the sample space, $\mathcal{F}$ is a σ-algebra on Ω that accounts for the universe of possible events, and $\mathcal{P}$ is a probability measure that assigns probabilities to events. In addition, the description establishes the measurable spaces $(\mathcal{X}, \mathcal{F}_t(\mathcal{X}))$ and $(\mathcal{Y}, \mathcal{F}_t(\mathcal{Y}))$ of the state and observation processes, respectively, with σ-algebras $\mathcal{F}_t(\mathcal{X}) \triangleq \sigma\{\mathrm{x}_t | t \geq 0\}$ and $\mathcal{F}_t(\mathcal{Y}) \triangleq \sigma\{\mathrm{y}_t | t > 0\}$ that compose the filtrations $\{\mathcal{F}_t(\mathcal{X})\}_{t\geq 0}$ and $\{\mathcal{F}_t(\mathcal{Y})\}_{t\geq 0}$ of the σ-algebra $\mathcal{F}$. The filtering problem concerns characterizing the system states by estimates conditioned on observations. In the optimal filtering formalism, the problem consists in finding the best $\mathbb{L}^2$-estimate of some $\mathcal{F}_t(\mathcal{X})$-measurable function of the state process, $\varphi(\mathrm{x}_t)$, based on the observations, that is, one wishes to find an $\mathcal{F}_t(\mathcal{Y})$-measurable quantity $\hat{\varphi}_t$ that minimizes $\|\hat{\varphi}_t - \varphi(\mathrm{x}_t)\|_2$. The formal solution to this problem is generally given by $\hat{\varphi}_t = \mathbb{E}\left[\varphi(\mathrm{x}_t)|\mathcal{F}_t(\mathcal{Y})\right]$ admitting almost sure equivalence, where $\mathbb{E}$ denotes the expectation with respect to the probability measure P.

In practice, observations are acquired at discrete time instants and so can be thought of as random sequences $\{\mathrm{y}_k\}_{k\in\mathbb{N}}$ realized at $t = 1\Delta t, \ldots, k\Delta t$. Similarly, as the state process shall be characterized conditioned on the observations, the states can also be denoted as discrete-time random sequences $\{\mathrm{x}_k\}_{k\in\mathbb{N}_0}$. Under these conditions, it is customary to presume that the system evolves according to a discrete-time Markov process, whose observations are conditionally independent given the states. In this setting, a standard Monte Carlo solution of the filtering problem is given in terms of N i.i.d. weighted random samples $\{w_k^{(i)}, \mathrm{x}_k^{(i)} | i = 1, \ldots, N\}$, such that

$$\sum_{i=1}^{N} w_k^{(i)} \varphi(\mathrm{x}_k^{(i)}) \overset{N\to\infty}{\longrightarrow} \int_{\mathcal{X}} \varphi(\mathrm{x}_k) p(\mathrm{x}_k|\mathrm{y}_1, \ldots, \mathrm{y}_k) d\mathrm{x}_k \quad \text{a.s.}, \tag{1.1}$$

where $p(\mathrm{x}_k|\mathrm{y}_1, \ldots, \mathrm{y}_k)$ is the posterior probability density of x_k conditioned on the history of observations. Hereafter the notations $\mathrm{x}_{0:k} \triangleq \{\mathrm{x}_0, \ldots, \mathrm{x}_k\}$ and $\mathrm{y}_{1:k} \triangleq \{\mathrm{y}_1, \ldots, \mathrm{y}_k\}$ shall be used.

1.2.2 Sequential Monte Carlo

In this and all subsequent sections, it is generally assumed that a Markov transition kernel $p(\mathrm{x}_k|\mathrm{x}_{k-1})$ and an observation likelihood function $p(\mathrm{y}_k|\mathrm{x}_k)$ are explicitly known or can be approximated to arbitrary accuracy, in the context of a hidden Markov model problem. The

perfect Monte Carlo solution to the filtering problem implies sampling directly from the target density $\pi(\mathrm{x}_k|\mathrm{y}_{1:k}) = p(\mathrm{x}_k|\mathrm{y}_1, \ldots, \mathrm{y}_k)$ to produce an equally weighted Monte Carlo distribution as

$$\hat{\pi}(\mathrm{x}_k|\mathrm{y}_{1:k}) = \frac{1}{N}\sum_{i=1}^{N} \delta(\mathrm{x}_k - \mathrm{x}_k^{(i)}), \quad \mathrm{x}_k^{(i)} \sim \pi(\mathrm{x}_k|\mathrm{y}_{1:k}), \tag{1.2}$$

where $\delta(\mathrm{x}_k - \mathrm{x}_k^{(i)})$ denotes a Dirac delta (probability) mass at $\mathrm{x}_k = \mathrm{x}_k^{(i)}$. This solution is rarely accessible because sampling from $\pi(\mathrm{x}_k|\mathrm{y}_{1:k})$ is often not possible. Instead, a more versatile solution relies on importance sampling (IS), where N i.i.d. random samples are taken from an importance density[2] $q(\mathrm{x}_k|\mathrm{y}_{1:k})$ which is "close" to the target density in some sense, and the mismatch between $q(\mathrm{x}_k|\mathrm{y}_{1:k})$ and $\pi(\mathrm{x}_k|\mathrm{y}_{1:k})$ is corrected by importance weights as

$$\begin{aligned}\sum_{i=1}^{N} w_k^{(i)} \varphi(\mathrm{x}_k^{(i)}) &\overset{N\to\infty}{\longrightarrow} \int_{\mathcal{X}} \varphi(\mathrm{x}_k)\pi(\mathrm{x}_k|\mathrm{y}_{1:k})d\mathrm{x}_k \\ &= \int_{\mathcal{X}} \underbrace{\frac{\pi(\mathrm{x}_k|\mathrm{y}_{1:k})}{q(\mathrm{x}_k|\mathrm{y}_{1:k})}}_{\breve{w}_k(\mathrm{x}_k|\mathrm{y}_{1:k})} \varphi(\mathrm{x}_k) q(\mathrm{x}_k|\mathrm{y}_{1:k}) d\mathrm{x}_k \quad \text{a.s.},\end{aligned} \tag{1.3}$$

where $w_k^{(i)} = \breve{w}_k(\mathrm{x}_k^{(i)}|\mathrm{y}_{1:k})/N$. This scheme produces the empirical Monte Carlo distribution

$$\hat{\pi}(\mathrm{x}_k|\mathrm{y}_{1:k}) = \sum_{i=1}^{N} w_k^{(i)} \delta(\mathrm{x}_k - \mathrm{x}_k^{(i)}), \quad \mathrm{x}_k^{(i)} \sim q(\mathrm{x}_k|\mathrm{y}_{1:k}). \tag{1.4}$$

In a sequential setting, this solution is more practical than the perfect Monte Carlo method. However, it requires evaluation of $\pi(\mathrm{x}_k|\mathrm{y}_{1:k})$ and sampling from $q(\mathrm{x}_k|\mathrm{y}_{1:k})$. Often, the evaluation of $\pi(\mathrm{x}_k|\mathrm{y}_{1:k})$ involves an approximation whose errors build up from previous steps [110], and $q(\mathrm{x}_k|\mathrm{y}_{1:k})$ must be constructed in the marginal space, i.e.

$$q(\mathrm{x}_k|\mathrm{y}_{1:k}) = \int_{\mathcal{X}} q(\mathrm{x}_k|\mathrm{x}_{k-1}, \mathrm{y}_k) q(\mathrm{x}_{k-1}|\mathrm{y}_{1:k-1}) d\mathrm{x}_{k-1},$$

which depends on the complete set of previous samples. Note that evaluating $\pi(\mathrm{x}_k|\mathrm{y}_{1:k})$ and constructing $q(\mathrm{x}_k|\mathrm{y}_{1:k})$ are bounded by $\mathcal{O}(N^2)$ operations. In contrast, standard particle filters establish a simpler framework, while demanding calculations bounded by only $\mathcal{O}(N)$ operations. Particle filters extend sample paths as $\mathrm{x}_{0:k}^{(i)} \equiv (\mathrm{x}_k^{(i)}, \mathrm{x}_{0:k-1}^{(i)})$ by sampling according to

$$\begin{aligned}\mathrm{x}_{0:k}^{(i)} &\sim q(\mathrm{x}_{0:k}|\mathrm{y}_{1:k}) = q(\mathrm{x}_k|\mathrm{x}_{0:k-1}, \mathrm{y}_k) q(\mathrm{x}_{0:k-1}|\mathrm{y}_{1:k-1}), \\ \mathrm{x}_k^{(i)} &\sim q(\mathrm{x}_k|\mathrm{x}_{0:k-1}^{(i)}, \mathrm{y}_k),\end{aligned} \tag{1.5}$$

and by performing importance sampling in the joint space of paths to estimate

$$\mathbb{E}_{q(\mathrm{x}_{0:k}|\mathrm{y}_{1:k})}\left[\breve{w}(x_{0:k}|y_{1:k})\varphi(\mathrm{x}_{0:k})\right] = \int_{\mathcal{X}^{k+1}} \underbrace{\frac{\pi(\mathrm{x}_{0:k}|\mathrm{y}_{1:k})}{q(\mathrm{x}_{0:k}|\mathrm{y}_{1:k})}}_{\breve{w}_k(x_{0:k}|y_{1:k})} \varphi(\mathrm{x}_{0:k}) q(\mathrm{x}_{0:k}|\mathrm{y}_{1:k}) d\mathrm{x}_{0:k}. \tag{1.6}$$

[2] Also known as proposal density.

Thus, the new weights are updated as

$$\begin{aligned}\breve{w}_k(\mathrm{x}_{0:k}|\mathrm{y}_{1:k}) &= \frac{\pi(\mathrm{x}_{0:k}|\mathrm{y}_{1:k})}{q(\mathrm{x}_{0:k}|\mathrm{y}_{1:k})} = \frac{\frac{p(\mathrm{y}_k|\mathrm{x}_k)p(\mathrm{x}_k|\mathrm{x}_{k-1})}{p(\mathrm{y}_k|\mathrm{y}_{1:k-1})}}{q(\mathrm{x}_k|\mathrm{x}_{0:k-1},\mathrm{y}_k)}\frac{\pi(\mathrm{x}_{0:k-1}|\mathrm{y}_{1:k-1})}{q(\mathrm{x}_{0:k-1}|\mathrm{y}_{1:k-1})} \\ &= \frac{1}{p(\mathrm{y}_k|\mathrm{y}_{1:k-1})}\frac{p(\mathrm{y}_k|\mathrm{x}_k)p(\mathrm{x}_k|\mathrm{x}_{k-1})}{q(\mathrm{x}_k|\mathrm{x}_{0:k-1},\mathrm{y}_k)}\breve{w}_{k-1}(\mathrm{x}_{0:k-1}|\mathrm{y}_{1:k-1}) \\ &\propto \frac{p(\mathrm{y}_k|\mathrm{x}_k)p(\mathrm{x}_k|\mathrm{x}_{k-1})}{q(\mathrm{x}_k|\mathrm{x}_{0:k-1},\mathrm{y}_k)}\breve{w}_{k-1}(\mathrm{x}_{0:k-1}|\mathrm{y}_{1:k-1}), \end{aligned} \tag{1.7}$$

where $p(\mathrm{x}_k|\mathrm{x}_{k-1})$ is the Markov transition kernel, $p(\mathrm{y}_k|\mathrm{x}_k)$ is the observation likelihood function and

$$p(\mathrm{y}_k|\mathrm{y}_{1:k-1}) = \int_{\mathcal{X}^{k+1}} p(\mathrm{y}_k|\mathrm{x}_k)p(\mathrm{x}_k|\mathrm{x}_{k-1})\pi(\mathrm{x}_{0:k-1}|\mathrm{y}_{1:k-1})d\mathrm{x}_{0:k}. \tag{1.8}$$

The normalizing constant (1.8) is inconvenient but can be eliminated by taking estimates as

$$\begin{aligned}\frac{\mathbb{E}_{q(\mathrm{x}_{0:k}|\mathrm{y}_{1:k})}\left[\breve{w}_k(\mathrm{x}_{0:k}|\mathrm{y}_{1:k})\varphi(\mathrm{x}_{0:k})\right]}{\mathbb{E}_{q(\mathrm{x}_{0:k}|\mathrm{y}_{1:k})}\left[\breve{w}_k(\mathrm{x}_{0:k}|\mathrm{y}_{1:k})\right]} &\approx \frac{\frac{1}{N}\sum_{i=1}^N \breve{w}_k^{(i)}\varphi(\mathrm{x}_{0:k}^{(i)})}{\frac{1}{N}\sum_{i=1}^N \breve{w}_k^{(i)}}, \\ &= \sum_{i=1}^N w_k^{(i)}\varphi(\mathrm{x}_k^{(i)}), \end{aligned} \tag{1.9}$$

where $\breve{w}_k^{(i)} = \breve{w}_k(\mathrm{x}_{0:k}^{(i)}|\mathrm{y}_{1:k})$ and so $w_k^{(i)} = \breve{w}_k^{(i)}/\sum_{i=1}^N \breve{w}_k^{(i)}$. Note that (1.5) is what causes the particle degeneracy, i.e. the extended sample paths inhabit a space with augmenting dimension as $\mathrm{x}_{0:k}^{(i)} \equiv (\mathrm{x}_k^{(i)}, \mathrm{x}_{0:k-1}^{(i)}) \in (\mathcal{X}, \mathcal{X}^k) = \mathcal{X}^{k+1}$, which makes them increasingly prone to the curse of dimensionality. In contrast, the filter proposed in this chapter constructs empirical measures in the marginal space akin to (1.2) by completely avoiding sequential importance sampling.

1.3 Grid-based posterior density approximation

Presuming that one possesses a previous set of samples $\{\mathrm{x}_{k-1}^{(i)}\}_{i\in[1..N]}$ distributed according to $\mathrm{x}_{k-1}^{(i)} \sim \hat{\pi}(\mathrm{x}_{k-1}|\mathrm{y}_{1:k-1})$, an empirical approximation to the Chapman-Komogorov equation can be achieved by

$$\begin{aligned} p(\mathrm{x}_k|\mathrm{y}_{1:k-1}) &= \int_{\mathcal{X}} p(\mathrm{x}_k|\mathrm{x}_{k-1})\pi(\mathrm{x}_{k-1}|\mathrm{y}_{1:k-1})d\mathrm{x}_{k-1}, \\ \hat{p}(\mathrm{x}_k|\mathrm{y}_{1:k-1}) &= \frac{1}{N}\sum_{i=1}^N p(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)}). \end{aligned} \tag{1.10}$$

Equation (1.10) is a mixture of probability densities given by a sum of transition kernels $p(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)})$ (continuous in x_k) over all particles $\{\mathrm{x}_{k-1}^{(i)}\}_{i\in[1..N]}$. This expression can also be understood as a sum of N realizations of local samplers, to give N probability distributions that jointly approximate a marginal distribution. In order to proceed with a discretized representation of $\hat{\pi}(\mathrm{x}_k|\mathrm{y}_{1:k-1})$, we note that it is possible to evaluate the mixture of densities at

abscissas of interest $\{\breve{\mathrm{x}}_k^{(a)}\}_{a\in[1..M^{d_x}]}$, to produce the probability mass function:

$$\begin{aligned} p(\mathrm{x}_k|\mathrm{y}_{1:k-1}) &\approx \hat{p}_{\mathrm{g}}(\mathrm{x}_k|\mathrm{y}_{1:k-1}) \\ &= \frac{1}{M}\sum_{a=1}^{M^{d_x}}\left(\frac{1}{N}\sum_{i=1}^{N} p(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)})\right)\delta(\mathrm{x}_k-\breve{\mathrm{x}}_k^{(a)}), \end{aligned} \tag{1.11}$$

where $M^{d_x} = M\times\cdots\times M$ (d_x times) nodes are considered. The discretized density (1.11) approximates the predicted density at $\{\breve{\mathrm{x}}_k^{(a)}\}_{a\in[1..M^{d_x}]}$. It could also be understood in terms of a kernel-based density estimate, where the kernel is $p(\mathrm{x}_k|\mathrm{x}_{k-1})$. An adequate discretized density can either be generated by evaluating (1.11) or by interpolating samples $\mathrm{x}_k^{(i)}\sim p(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)})$ at the points $\{\breve{\mathrm{x}}_k^{(a)}\}_{a\in[1..M^{d_x}]}$.

It is convenient to set $M \ll N$, i.e. the number of points of interest per dimension to discretize the posterior density should be much smaller than the number of particles. This property is relevant to get a computationally efficient estimation algorithm, by allowing a minimization of M to limit the number of mixture evaluations. The method advocated in this chapter is to evaluate the sums (1.11) by setting $\{\breve{\mathrm{x}}_k^{(a)}\}_{a\in[1..M^{d_x}]}$ in terms of the roots of Mth-order Hermite polynomials, envisaging to perform the Gauss-Hermite quadrature according to the technique that will be explained in Section 1.5. The number of computations can be effectively reduced by taking into account only the components $p(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)})$ for which $\{\mathrm{x}_{k-1}^{(i)}\}$ fall within the neighborhood of each $\breve{\mathrm{x}}_k^{(a)}$, or by considering the k nearest neighbors $\{\mathrm{x}_{k-1}^{(i)}\}$ of $\breve{\mathrm{x}}_k^{(a)}$, which can be found in $\mathcal{O}(\log N)$ using spatial data structures (e.g., kd-trees).

Finally, the induction of the posterior density approximation can be achieved by performing a pointwise Bayesian update. A straightforward approximation to the Bayes' rule can be acquired by evaluating $p(\mathrm{y}_k|\mathrm{x}_k)$ at the current observation y_k given the points $\breve{\mathrm{x}}_k^{(a)}$, and proceeding the multiplication to get

$$\begin{aligned} \pi(\mathrm{x}_k|\mathrm{y}_{1:k}) &\approx \hat{\pi}_{\mathrm{g}}(\mathrm{x}_k|\mathrm{y}_{1:k}) \\ &= \frac{\sum_{a=1}^{M^{d_x}} p(\mathrm{y}_k|\mathrm{x}_k)\left(\sum_{i=1}^{N} p(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)})\right)\delta(\mathrm{x}_k-\breve{\mathrm{x}}_k^{(a)})}{\sum_{a'=1}^{M^{d_x}} p(\mathrm{y}_k|\breve{\mathrm{x}}_k^{(a')})\left(\sum_{i=1}^{N} p(\breve{\mathrm{x}}_k^{(a')}|\mathrm{x}_{k-1}^{(i)})\right)}. \end{aligned} \tag{1.12}$$

The posterior probability distribution approximated by (1.12) enables estimates as

$$\mathbb{E}_{\pi}\left[\varphi(\mathrm{x}_k)|\mathrm{y}_{1:k}\right] \approx \frac{\sum_{a=1}^{M^{d_x}} p(\mathrm{y}_k|\breve{\mathrm{x}}_k^{(a)})\left(\sum_{i=1}^{N} p(\breve{\mathrm{x}}_k^{(a)}|\mathrm{x}_{k-1}^{(i)})\right)\varphi(\breve{\mathrm{x}}_k^{(a)})}{\sum_{a'=1}^{M^{d_x}} p(\mathrm{y}_k|\breve{\mathrm{x}}_k^{(a')})\left(\sum_{i=1}^{N} p(\breve{\mathrm{x}}_k^{(a')}|\mathrm{x}_{k-1}^{(i)})\right)}. \tag{1.13}$$

The advantage of this estimation method will become clear later on: the idea is that the probability masses expressed in the grid consolidate the uncertainty from Monte Carlo samples with an inherent regularizing characteristic. This is achieved because a grid-based probability mass function diminishes the effect of outliers in the estimates and smoothes the posterior representation due to the averaging mechanism (over samples) required to compute the masses' weights. In addition, the Monte Carlo scheme works as a complementary feature, being responsible for propagating the state uncertainty by particles and so capitalizing on a concise and sparse representation of the distributions involved.

1.4 Particles redrawing

In the procedure described in Section 1.3, a previous set of samples $\{\mathrm{x}_{k-1}^{(i)}\}_{i\in[1..N]}$ is used to induce a discretized representation of the posterior distribution on a grid. In order to complete the recursion, the approximated posterior measure should characterize the distribution of a set of posterior samples $\{\mathrm{x}_k^{(i)}\}_{i\in[1..N]}$, i.e, $\mathrm{x}_k^{(i)} \sim \hat{\pi}_{\mathrm{g}}(\mathrm{x}_k|\mathrm{y}_{1:k})$. To do so one should be able to draw samples from $\hat{\pi}_{\mathrm{g}}(\mathrm{x}_k|\mathrm{y}_{1:k})$, which is not straightforward. This section presents some methods for propagating samples so that they become distributed according to the empirical posterior distribution, and hence close the recursion for the proposed filter.

1.4.1 Sampling

A particularly useful interpretation of the posterior involves updating the predicted distribution (1.10) via Bayes' rule as

$$\begin{aligned}\hat{\pi}(\mathrm{x}_k|\mathrm{y}_{1:k}) &= \frac{p(\mathrm{y}_k|\mathrm{x}_k)\hat{p}(\mathrm{x}_k|\mathrm{y}_{1:k-1})}{p(\mathrm{y}_k|\mathrm{y}_{1:k-1})} = \frac{p(\mathrm{y}_k|\mathrm{x}_k)N^{-1}\sum_{i=1}^{N} p(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)})}{p(\mathrm{y}_k|\mathrm{y}_{1:k-1})} \\ &= \sum_{i=1}^{N} w_{\pi,k}^{(i)} \frac{p(\mathrm{y}_k|\mathrm{x}_k)p(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)})}{p(\mathrm{y}_k|\mathrm{x}_{k-1}^{(i)})} = \sum_{i=1}^{N} w_{\pi,k}^{(i)} p(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)}, \mathrm{y}_k) \\ &\approx \sum_{i=1}^{N} \hat{w}_{\pi,k}^{(i)} \mathcal{N}(\mathrm{x}_k; \mathrm{m}_k^{(i)}, \Sigma_k^{(i)}),\end{aligned} \tag{1.14}$$

where the Markov transition kernel and the likelihood function are assumed to be well approximated by Gaussian density functions as $p(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)}) \approx \mathcal{N}(\mathrm{x}_k; f(\mathrm{x}_{k-1}^{(i)}), \mathrm{Q}_k^{(i)})$ and $p(\mathrm{y}_k|\mathrm{x}_k) \approx \mathcal{N}(\mathrm{y}_k; h(\mathrm{x}_k), \mathrm{R}_k)$, respectively, and the statistics of each mixture component are given by [64]

$$\Sigma_k^{(i)} = (\mathrm{Q}_k^{(i)\,-1} + \mathrm{H}^{(i)\,T}\mathrm{R}_k^{-1}\mathrm{H}^{(i)})^{-1}, \tag{1.15}$$

$$\mathrm{m}_k^{(i)} = \Sigma_k^{(i)}(\mathrm{Q}_k^{(i)\,-1} f(\mathrm{x}_{k-1}^{(i)}) + \mathrm{H}^{(i)\,T}\mathrm{R}_k^{-1}\mathrm{y}_k), \tag{1.16}$$

where $f : \mathcal{X} \to \mathcal{X}$ is the state transition function, $\mathrm{H}^{(i)}$ is an observation matrix that approximates an observation function $h : \mathcal{X} \to \mathcal{Y}$ as a linear operator (e.g., in the EKF or UKF sense), $\mathrm{Q}_k^{(i)}$ is the process noise covariance matrix, and R_k is the observation noise covariance matrix. In addition,

$$\begin{aligned} w_{\pi,k}^{(i)} &\triangleq \frac{p(\mathrm{y}_k|\mathrm{x}_{k-1}^{(i)})}{N \cdot p(\mathrm{y}_k|\mathrm{y}_{1:k-1})} \\ &\propto p(\mathrm{y}_k|\mathrm{x}_{k-1}^{(i)}) = \int_{\mathcal{X}} p(\mathrm{y}_k|\mathrm{x}_k)p(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)})d\mathrm{x}_k, \\ \hat{w}_{\pi,k}^{(i)} &\approx \frac{\mathcal{N}(\mathrm{y}_k; \mathrm{H}^{(i)} f(\mathrm{x}_{k-1}^{(i)}), \mathrm{H}^{(i)}\mathrm{Q}_k^{(i)}\mathrm{H}^{(i)\,T} + \mathrm{R}_k)}{\sum_{j=1}^{N} \mathcal{N}(\mathrm{y}_k; \mathrm{H}^{(j)} f(\mathrm{x}_{k-1}^{(j)}), \mathrm{H}^{(j)}\mathrm{Q}_k^{(j)}\mathrm{H}^{(j)\,T} + \mathrm{R}_k)}. \end{aligned} \tag{1.17}$$

Note that the factor $(N \cdot p(\mathrm{y}_k|\mathrm{y}_{1:k-1}))^{-1}$ weights all components equally in (1.14) and so does not appear in (1.17). Matrices and weights indexed by $i \in [1..N]$ denote per-sample (local) approximations. In this interpretation, sampling from $\hat{\pi}(\mathrm{x}_k|\mathrm{y}_{1:k})$ just requires taking samples from the components $p(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)}, \mathrm{y}_k)$ in (1.14) with probability $w_{\pi,k}^{(i)}$. This is actually easy

to achieve as shown in *Algorithm 1.1*. Also, note that, in the step 7 of this algorithm, one could use the selected component mean to directly represent the new probability mass, i.e. $\mathrm{x}_k^{(i)} \leftarrow \mathrm{m}_k^{(j)}$. For simple filtering cases, this seems justifiable, however, the mixture component mean consolidates the local uncertainty in a smoothing statistic that, for fast mixing Markov chains, would inhibit the complete exploration of the target measure and would lead to the impoverishment of particles over time. This problem is intimately related to that reported for Gaussian sum particle filters as the *collapsing of mixands* [112], and can be seen as a variation of the sample impoverishment observed in deterministic resampling algorithms for which particles' regularization [145] is proposed as a solution. By opting for sampling in step 7 of *Algorithm 1.1*, the particles' regularization is systematically achieved.

Various other methods for treating posterior modes separately are readily available for grid-based filters (e.g., [21, 178]), but are less general than that presented in this section. It is worth noting that the recursion established for the hybrid Gauss-Hermite filter resembles the Gaussian particle filter [111] in the sense that it parameterizes an approximation of the posterior density to perform the redrawing step when necessary.

In some specific cases, a single marginal proposal distribution can be constructed according to [195], when the posterior distribution can be well approximated as a single Gaussian via the UKF (or EKF). This is done by applying one instance of the UKF (or EKF) to the whole population of previous samples. The feasibility of the method proposed in [195] depends on a number of factors, such as the complexity of the state process, number of relevant modes in the posterior density, desired computational efficiency, and whether the resulting errors are reasonable.

Algorithm 1.1: Sampling from the approximate posterior mixture

Input : Previous set of particles $\{\mathrm{x}_{k-1}^{(i)}\}_{i\in[1..N]}$, observation y_k

1 Compute posterior mixture weights:

$$\hat{w}_{\pi,k}^{(i)} = \frac{\mathcal{N}(\mathrm{y}_k; \mathrm{H}^{(i)} f(\mathrm{x}_{k-1}^{(i)}), \mathrm{H}^{(i)} \mathrm{Q}_k^{(i)} \mathrm{H}^{(i)\,T} + \mathrm{R}_k)}{\sum_{j=1}^{N} \mathcal{N}(\mathrm{y}_k; \mathrm{H}^{(j)} f(\mathrm{x}_{k-1}^{(j)}), \mathrm{H}^{(j)} \mathrm{Q}_k^{(j)} \mathrm{H}^{(j)\,T} + \mathrm{R}_k)}$$

2 Calculate cumulative sum function of the weights: $c(0) \coloneqq 0$, $c(n) = \sum_{i=1}^{n} \hat{w}_{\pi,k}^{(i)}$ for $n \in [1..N]$

3 **Draw samples from the posterior mixture:**

4 **for** $i = 1, \ldots, N$ **do**

5 Draw a uniformly distributed test variable: $u^{(i)} \sim \mathcal{U}([0,1])$

6 Find the inverse of the cumulative sum function: $j \coloneqq c^{-1}(u^{(i)})$ such that $u^{(i)} \in [c(j-1), c(j))$

7 Sample a new ith particle from the jth component:

$$\mathrm{x}_k^{(i)} \sim \hat{p}(\mathrm{x}_k | \mathrm{x}_{k-1}^{(j)}, \mathrm{y}_k) = \mathcal{N}(\mathrm{x}_k; \mathrm{m}_k^{(j)}, \Sigma_k^{(j)})$$

8 **end**

Output: New set of particles $\{\mathrm{x}_k^{(i)}\}_{i\in[1..N]}$

1.4.2 Ensuring convergence

In complex cases, when highly non-Gaussian posterior distributions are present, sampling from an approximation of $\hat{\pi}(\mathrm{x}_k|\mathrm{y}_{1:k})$ may not guarantee that the posterior distribution is being correctly explored by the samples. Ensuring convergence to a stationary target measure is usually achieved in Markov chain Monte Carlo (MCMC) methods by introducing an accept-reject test. This accept-reject test is set to discard moves of the Markov chain that are unlikely to happen when exploring a stationary target measure, in the context of the Metropolis-Hastings algorithm. The same principle can be used for sampling from $\hat{\pi}(\mathrm{x}_k|\mathrm{y}_{1:k})$, such that one can propose moves based on the method of Section 1.4.1, i.e., propose a candidate

$$\mathrm{x}_k^{\star(i)} \sim q(\mathrm{x}_k|\mathrm{x}_{k-1}^{(j)}, \mathrm{y}_k), \tag{1.18}$$

with probability $w_{\pi,k}^{(j)}$, and accept it with acceptance probability

$$\alpha^{(i)} = \min\left(\frac{p(\mathrm{y}_k|\mathrm{x}_k^{\star(i)})\hat{p}(\mathrm{x}_k^{\star(i)}|\mathrm{y}_{1:k-1})}{p(\mathrm{y}_k|\mathrm{x}_{k-1}^{(j)})\hat{p}(\mathrm{x}_{k-1}^{(j)}|\mathrm{y}_{1:k-1})}\frac{q(\mathrm{x}_{k-1}^{(j)}|\mathrm{x}_k^{\star(i)}, \mathrm{y}_k)}{q(\mathrm{x}_k^{\star(i)}|\mathrm{x}_{k-1}^{(j)}, \mathrm{y}_k)}, 1\right). \tag{1.19}$$

By drawing a uniformly distributed test variable $u^{(i)} \sim \mathcal{U}([0,1])$, if $u^{(i)} \leq \alpha^{(i)}$ then the move is accepted as $\mathrm{x}_k^{(i)} \leftarrow \mathrm{x}_k^{*(i)}$, else rejected as $\mathrm{x}_k^{(i)} \leftarrow \mathrm{x}_{k-1}^{(j)}$.

1.5 Gauss-Hermite quadrature

Gauss-Hermite quadrature is an accurate method for numerical integration, applicable to many fields of Physics and Engineering. It is specially useful in statistics due to the close relationship with Gaussian densities, but it is generally difficult to be applied to problems with sparse data. The Gauss-Hermite quadrature proposes a numerical solution to integrals of the form

$$\int_{-\infty}^{+\infty} \varphi(x)\, e^{-x^2} dx \approx \sum_{j=1}^{M} \bar{w}_j \varphi(x_j), \tag{1.20}$$

where the nodes or abscissae $\{x_j\}_{j\in\mathbb{N}_0}$ are the roots of an M*th*-order Hermite polynomial and $\{\bar{w}_j\}_{j\in\mathbb{N}_0}$ are the corresponding quadrature weights. The abscissae are symmetric about zero, discretizing the function $\varphi(x)$ at points where the integral is expected to be efficiently approximated by a relatively small number of components. The quadrature weights are acquired by interpreting the integral as an inner product of orthogonal polynomials with respect to e^{-x^2}, thus

$$\bar{w}_j = \frac{\langle H_{M-1}, H_{M-1}\rangle_{e^{-x^2}}}{H_{M-1}(x_j)H'_{M-1}(x_j)} = \frac{2^{M-1}M!\sqrt{\pi}}{M^2 H_{M-1}(x_j)^2}, \tag{1.21}$$

where H_M is a Hermite polynomial of the order M and H'_{M-1} is its derivative with respect to x. Zeros of Hermite polynomials and their corresponding weights can be found tabulated in many references (e.g., [189]) up to a certain order and at a certain numerical precision. In order to obtain zeros of a M*th*-order Hermite polynomial and their weights, we suggest the recursive procedure proposed by [157] to avoid the evaluation of factorials in (1.21). The quadrature

weights can be calculated by

$$\bar{w}_j = \frac{2}{(\breve{H}'_j)^2}, \tag{1.22}$$

where $\{\breve{H}_j\}_{j\in\mathbb{N}_0}$ is an orthonormal set of polynomials generated by the recurrence

$$\begin{aligned}
\breve{H}_{-1} &= 0, \\
\breve{H}_0 &= \frac{1}{\pi^{1/4}}, \\
&\vdots \\
\breve{H}_{j+1} &= x\sqrt{\frac{2}{j+1}}\breve{H}_j - \sqrt{\frac{j}{j+1}}\breve{H}_{j-1},
\end{aligned} \tag{1.23}$$

and the derivative of the normalized polynomials can be achieved by $\breve{H}'_j = \sqrt{2j}\breve{H}_{j-1}$. The roots of a polynomial $\breve{H}_j$ can obtained by the Newton-Raphson method.

Theoretically, integrating by the Gauss-Hermite quadrature rule provides an exact result for polynomial integrands of the order up to $2M-1$ [189]. The main difficulty in applying the GHQ lies in the fact that the function $\varphi(x)$ may require discretization at points irrespective of the range where the e^{-x^2} has substantial values. Therefore, in order to obtain accurate results, a transformation must be made so that the integrand is represented in a convenient range, covering the region where the integrand is most significant. Another concern is that it is useful to evaluate the integral for any continuous function, not only integrands of the form $\varphi(x)\,e^{-x^2}$. Thus one shall consider a systematic procedure, as presented in [124], to apply the GHQ to integrals of the form

$$\int_{-\infty}^{+\infty} \psi(x)\,dx = \int_{-\infty}^{+\infty} \varphi(z)\,\mathcal{N}(z;\mu_z,\sigma_z)dz, \tag{1.24}$$

In order to get bounded errors in the numerical integration, the ratio of $\psi(x)$ to some Gaussian curve shall be a moderately smooth function [124]. This is true for many problems in statistics where, for example, $\psi(x)$ is a product of several likelihood functions, or a product of likelihood functions and other exponential densities. A transformation on x is proposed so that the integrand $\psi(x)$ will be discretized in a suitable range for an accurate quadrature. Thus, the nodes $\{x_j\}_{j\in[1..M]}$, located at the roots of an Mth-order Hermite polynomial, are then transferred and scaled into a new set of nodes $\{z_a\}_{a\in[1..M]}$, by the linear transformation $z_a = \mu_z + \sqrt{2}\sigma_z \cdot x_j$, with weights given by $\bar{w}_a = \bar{w}_j/\sqrt{\pi}$. Following this procedure, one obtains

$$\begin{aligned}
\int_{-\infty}^{+\infty} \varphi(z)\mathcal{N}(z;\mu_z,\sigma_z)dz &\approx \sum_{a=1}^{M} \bar{w}_a\varphi(z_a) \\
&= \sum_{j=1}^{M} \frac{\bar{w}_j}{\sqrt{\pi}}\varphi(\mu_z + \sqrt{2}\sigma_z \cdot x_j) \\
&= \sum_{j=1}^{M} \frac{\bar{w}_j}{\sqrt{\pi}}\frac{\psi(\mu_z + \sqrt{2}\sigma_z \cdot x_j)}{\mathcal{N}(z;\mu_z,\sigma_z)} \\
&= \sqrt{2}\sigma_z \sum_{j=1}^{M} \bar{w}_j \cdot e^{x_j^2}\psi(\mu_z + \sqrt{2}\sigma_z \cdot x_j),
\end{aligned}$$

$$\therefore \int_{-\infty}^{+\infty} \psi(x)\, dx \approx \hat{\sigma}_z \sum_{j=1}^{M} \bar{w}_j \cdot e^{x_j^2} \psi(\mu_z + \hat{\sigma}_z \cdot x_j), \tag{1.25}$$

where $\hat{\sigma}_z = \sigma_z\sqrt{2}$. For evaluating the integral (1.24) it is sufficient to know the original abscissae x_j (Hermite polynomial roots) around zero and their corresponding weights $\bar{w}_j$, so that one can apply a linear transformation to the abscissae of $\psi(x)$ for adjusting the scale and the position to where the integrand provides suitable values. The abscissae x_j and weights $\bar{w}_j$ just have to be calculated once, or be read from a table pre-calculated offline, and integrals of arbitrary functions can be evaluated by expression (1.25).

1.6 Posterior estimate

The method presented in this section enables calculation of the posterior state mean and the covariance matrix. In the Gauss-Hermite quadrature framework, the posterior probability density (1.12) can be written as

$$\begin{aligned}\hat{\pi}_{\mathrm{g}}(\mathrm{x}_k|\mathrm{y}_{1:k}) &= \frac{\sum_{a=1}^{M^{d_x}} \bar{w}_a \cdot p(\mathrm{y}_k|\mathrm{x}_k) \left(\sum_{i=1}^{N} p(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)})\right) \delta(\mathrm{x}_k - \breve{\mathrm{x}}_k^{(a)})}{\sum_{a'=1}^{M^{d_x}} \bar{w}_{a'} \cdot p(\mathrm{y}_k|\breve{\mathrm{x}}_k^{(a')}) \left(\sum_{i=1}^{N} p(\breve{\mathrm{x}}_k^{(a')}|\mathrm{x}_{k-1}^{(i)})\right)} \\ &= \sum_{a=1}^{M^{d_x}} \hat{w}_{\pi_{\mathrm{g}},k}^{(a)}\, \delta(\mathrm{x}_k - \breve{\mathrm{x}}_k^{(a)}),\end{aligned} \tag{1.26}$$

where

$$\hat{w}_{\pi_{\mathrm{g}},k}^{(a)} \triangleq \frac{\bar{w}_a \cdot p(\mathrm{y}_k|\breve{\mathrm{x}}_k^{(a)}) \left(\sum_{i=1}^{N} p(\breve{\mathrm{x}}_k^{(a)}|\mathrm{x}_{k-1}^{(i)})\right)}{\sum_{a'=1}^{M^{d_x}} \bar{w}_{a'} \cdot p(\mathrm{y}_k|\breve{\mathrm{x}}_k^{(a')}) \left(\sum_{i=1}^{N} p(\breve{\mathrm{x}}_k^{(a')}|\mathrm{x}_{k-1}^{(i)})\right)}, \tag{1.27}$$

and each node of the multidimensional grid, $(i_1, \ldots, i_{d_x}) \in [0..M-1]^{d_x}$, is indexed by a correspondence map $a(i_1, \ldots, i_{d_x})$, $a : [0..M-1]^{d_x} \rightarrow [1..M^{d_x}]$, $M^{d_x} = M \times \cdots \times M$. The multidimensional quadrature weights $\{\bar{w}_a\}_{a\in[1..M^{d_x}]}$ are given by

$$\bar{w}_a \propto \bar{w}_{i_1} \ldots \bar{w}_{i_{d_x}} \exp(x_{i_1}^2 + \cdots + x_{i_{d_x}}^2), \tag{1.28}$$

where $\{x_j\}_{j\in[0..M-1]}$ are the one-dimensional abscissas (polynomial roots), and $\{\bar{w}_j\}_{j\in[0..M-1]}$ are the original quadrature weights. By writing the multidimensional abscissae as $\mathrm{x}^{(a)} \coloneqq (x_{i_1}, \ldots, x_{i_{d_x}})^T$ for all $(i_1, \ldots, i_{d_x}) \in [0..M-1]^{d_x}$, the transformed abscissae are calculated as

$$\breve{\mathrm{x}}_k^{(a)} = \mu_{\mathrm{z},k} + \Sigma_{\mathrm{z},k}^{1/2} \cdot \mathrm{x}^{(a)}, \quad a = 1, \ldots, M^{d_x}. \tag{1.29}$$

The points $\{\breve{\mathrm{x}}_k^{(a)}\}_{a\in[1..M^{d_x}]}$ must be systematically defined to contain all points of interest of the posterior $\hat{\pi}(\mathrm{x}_k|\mathrm{y}_{1:k})$. The transformation (1.29) is defined in terms of the parameters $\mu_{\mathrm{z},k}$ and $\Sigma_{\mathrm{z},k}^{1/2}$ that may be interpreted as parameters of an arbitrary Gaussian density transformed from $\exp(-\mathrm{x}^{(a)\,T}\mathrm{x}^{(a)})$. However, in terms of the formulation presented herein, these parameters don't necessarily have a statistical meaning. One is just interested in scaling and positioning the abscissae where $\hat{\pi}(\mathrm{x}_k|\mathrm{y}_{1:k})$ encloses a high probability region, to enable numerical adequacy

for computing the moments. The moments of the filtered state can computed by

$$\bar{\mathrm{x}}_k = \mathbb{E}_\pi[\mathrm{x}_k] = \int_{\mathcal{X}} \mathrm{x}_k \hat{\pi}_{\mathrm{g}}(\mathrm{x}_k|\mathrm{y}_{1:k}) d\mathrm{x}_k \approx \sum_{a=1}^{M^{d_x}} \hat{w}_{\pi_{\mathrm{g}},k}^{(a)} \breve{\mathrm{x}}_k^{(a)}, \tag{1.30}$$

$$\mathrm{P}_{\mathrm{x},k} = \mathbb{E}_\pi[(\mathrm{x}_k - \bar{\mathrm{x}}_k)(\mathrm{x}_k - \bar{\mathrm{x}}_k)^T] \approx \sum_{a=1}^{M^{d_x}} \hat{w}_{\pi_{\mathrm{g}},k}^{(a)} (\breve{\mathrm{x}}_k^{(a)} - \bar{\mathrm{x}}_k)(\breve{\mathrm{x}}_k^{(a)} - \bar{\mathrm{x}}_k)^T, \tag{1.31}$$

For notation simplicity, (1.30) and (1.31) are expressed with a single sum over a total of M^{d_x} grid points for d_x dimensions. In multidimensional cases this sum could be split into d_x sums, one per each dimension, accounting for multiple one-dimensional quadrature rules [99]. It is worth noting that the algorithm complexity is exponential on the number of dimensions and, hence, it is suitable for low-dimensional nonlinear problems that either require high estimation accuracy or are difficult to be addressed by either the EKF, UKF, or standard particle filters. A particle filter with N particles performs computations bounded by $\mathcal{O}(N)$ operations per iteration, a standard grid-based filter with N cells has complexity bounded by $\mathcal{O}(N^2)$ operations[3] per iteration [4], whereas the hybrid Gauss-Hermite filter with N particles and M nodes per dimension, has complexity bounded by $\mathcal{O}(NM^{d_x})$ operations per cycle. Therefore, the feasibility of the filter must be analyzed based on a trade-off between accuracy and complexity, which is exercised by properly choosing the minimum number of particles N and nodes per dimension M to solve a given estimation problem to a desired accuracy. For a simple treatment of the considerations to be taken when performing a multidimensional GHQ, the reader is referred to [99].

1.7 Grid positioning

Established techniques for grid-based estimators (e.g, [21, 178]) could be used to set the state-space range covered by the nodes $\{\breve{\mathrm{x}}_k^{(a)}\}_{a\in[1..M^{d_x}]}$. In principle, the state-space range covered by the grid should accommodate the support of the posterior density function (or the sub-domain where the function evaluations are numerically significant). Also, it is important to note that, ideally, the spacing between points in the grid should be decided upon the ability of the grid to represent, in the state space, all the relevant spectral content of the posterior probability density function. This becomes clear when we observe that an increasingly "peaky" posterior density (e.g., a Gaussian with very small variance) would require decreasingly small grid spacings. The rule to be employed should be analogous to the Nyquist criterion for discretizing time-series signals. That is, along each dimension, the maximum spacing between grid points should be smaller than a fraction ($1/2$ in the Nyquist criterion) of the spatial constant[4] of the highest-frequency spectral component that is relevant to describe the posterior density, i.e. the spacing between grid points should be inversely proportional to the posterior density bandwidth. This criterion, in the quadrature context, is directly related to the result that a grid with M nodes would produce exact integration for polynomial integrands of the order up to $2M-1$ [189]. Ultimately, the ideal solution would impose a grid with a total number of nodes adaptively determined, where the maximum spacing between nodes should be determined

[3] Each operation assumed on $\mathrm{x}_t \in \mathbb{R}^{d_x}$.

[4] Analogous of the time constant for time-series signals.

based on estimated higher-order moments derived from the sample population.

Nevertheless, the filter proposed in this chapter adopts a grid with fixed size and a simple technique to define the grid range in terms of redrawn samples. The goal is to compute the parameters for the transformation (1.29) as follows. From the population of redrawn particles $\{\mathrm{x}_k^{(i)}\}_{i\in[1..N]}$, one calculates the sample mean $\bar{\mathrm{x}}_{S,k}$ and empirical covariance $\mathrm{P}_{S,k}$, and then computes the transformation parameters $\mu_{\mathrm{z},k}$ and $\Sigma_{\mathrm{z},k}^{1/2}$ by

$$\bar{\mathrm{x}}_{S,k} = N^{-1}\sum_{i=1}^{N}\mathrm{x}_k^{(i)}, \quad \mathrm{P}_{S,k} = N^{-1}\sum_{i=1}^{N}(\mathrm{x}_k^{(i)} - \bar{\mathrm{x}}_{S,k})(\mathrm{x}_k^{(i)} - \bar{\mathrm{x}}_{S,k})^T, \tag{1.32}$$

$$\mu_{\mathrm{z},k} = \bar{\mathrm{x}}_{S,k}, \quad \Sigma_{\mathrm{z},k}^{1/2} = \ell \cdot \mathrm{P}_{S,k}^{1/2}/\|\Delta\mathrm{x}^{(a)}\|, \tag{1.33}$$

where ℓ is a scaling factor[5] and $\|\Delta\mathrm{x}^{(a)}\|$ is the Euclidean norm of the range covered by the basic set of (multidimensional) abscissae $\{\mathrm{x}^{(a)}\}_{j\in[1..M^{d_x}]}$. In (1.33), $\mathrm{P}_{S,k}^{1/2}$ is the lower triangular matrix obtained by the Cholesky decomposition of $\mathrm{P}_{S,k}$. Depending on the symmetry of the expected posterior distribution, it may be convenient to rotate the grid in order to align the principal axes of the sample covariance ellipsoid with the edges of the d_x-cube grid. This procedure consists in finding an eigensystem $\mathrm{V}_{S,k}$ that diagonalizes $\mathrm{P}_{S,k}$, along with its diagonal eigenvalue matrix $\Lambda_{S,k}$, for calculating an affine-transformed matrix $\mathrm{D}_{S,k}^{1/2} = \mathrm{V}_{S,k}\cdot\sqrt{\Lambda_{S,k}}$ that substitutes $\mathrm{P}_{S,k}^{1/2}$ in the formula (1.33). If desirable, by an additional rotation, it is possible to align the principal axes of the sample covariance ellipsoid with the diagonals of the grid hypercube.

During the redrawing step, if a single marginal proposal distribution is constructed according to [195], then finding the positioning and scaling parameters can be directly based on the moments of the approximating Gaussian density. As in this case the proposal is chosen to describe the whole population of particles, i.e. applying one instance of EKF/UKF for all particles, its moments can be used to calculate the parameters of the transformation (1.29) according to equations (1.33) by substituting $(\bar{\mathrm{x}}_{S,k}, \mathrm{P}_{S,k}^{1/2})$ with the proposal moments.

It is remarkable that, despite that defining a suitable grid area is essential to render a good algorithm performance, the numerical integration accuracy is weakly dependent on deviations of the position and scale of the grid. This is due to the nature of the Gauss-Hermite quadrature, whose accuracy relies on the orthogonal base of its terms rather than on the precision of points where the integrand function is evaluated. This property is analogous, and closely related, to the relatively low sensitivity of a Fourier transform of a smooth signal, to small changes in the period of analysis. Provided that the grid is not completely misplaced, i.e. on a region that is not covered by the support of the integrand function, the algorithm's performance is not sensible to changes in the position of nodes. The hybrid Gauss-Hermite algorithm is outlined in *Algorithm 1.2*. Note that there is no resample step in the proposed algorithm.

1.8 Examples and results

In this section we consider two simulation examples to illustrate the application of the hybrid Gauss-Hermite filter. The first example is a unidimensional nonlinear process with nonlinear observation model that is widely studied in the particle filtering literature. The second example

[5] As rule of thumb, $\ell = 3$ is recommended for smooth posterior distributions.

Algorithm 1.2: Hybrid Gauss-Hermite filter

Initialization:

if *time* $k = 0$ **then**

Sample $\mathrm{x}_0^{(i)} \sim \pi(\mathrm{x}_0)$ for $i = 1, \ldots, N$

Compute or read the original abscissae $\{\mathrm{x}^{(a)}\}_{j \in [1..M^{d_x}]}$ and weights $\{\bar{w}_a\}_{a \in [1..M^{d_x}]}$ for an Mth-order Gauss-Hermite quadrature

/* By fixing the grid size, the algorithm assumes that the target densities can be accurately described by polynomials of the order up to $2M-1$ (along each dimension). However, as touched on before, the ideal solution would require a grid size adaptively determined, where the maximum spacing between nodes should be determined based on higher-order moments estimated from the sample population. */

end

Steps:

for *time* $k \geq 1$ **do**

Sample new set of particles $\{\mathrm{x}_k^{(i)}\}_{i \in [1..N]}$ (*Algorithm 1.1*) as

$$\mathrm{x}_k^{(i)} \sim \hat{\pi}(\mathrm{x}_k|\mathrm{y}_{1:k}) = \sum_{i=1}^{N} \hat{w}_{\pi,k}^{(i)}\, \hat{p}(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)}, \mathrm{y}_k)$$

Compute positioning and scaling parameters:

$$\bar{\mathrm{x}}_{S,k} = N^{-1} \sum_{i=1}^{N} \mathrm{x}_k^{(i)}, \quad \mathrm{P}_{S,k} = N^{-1} \sum_{i=1}^{N} (\mathrm{x}_k^{(i)} - \bar{\mathrm{x}}_{S,k})(\mathrm{x}_k^{(i)} - \bar{\mathrm{x}}_{S,k})^T,$$
$$\mu_{\mathrm{z},k} = \bar{\mathrm{x}}_{S,k}, \quad \Sigma_{\mathrm{z},k}^{\frac{1}{2}} = \ell \cdot \mathrm{P}_{S,k}^{1/2} / \|\Delta \mathrm{x}^{(a)}\|$$

Position and scale the quadrature grid:

$$\breve{\mathrm{x}}_k^{(a)} = \mu_{\mathrm{z},k} + \Sigma_{\mathrm{z},k}^{1/2} \cdot \mathrm{x}^{(a)}, \quad a = 1, \ldots, M^{d_x}$$

Output: Approximation of the quadrature-based filtering distribution

$$\hat{\pi}_{\mathrm{g}}(\mathrm{x}_k|\mathrm{y}_{1:k}) = \frac{\sum_{a=1}^{M^{d_x}} \bar{w}_a \cdot p(\mathrm{y}_k|\mathrm{x}_k) \left(\sum_{i=1}^{N} p(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)})\right) \delta(\mathrm{x}_k - \breve{\mathrm{x}}_k^{(a)})}{\sum_{a'=1}^{M^{d_x}} \bar{w}_{a'} \cdot p(\mathrm{y}_k|\breve{\mathrm{x}}_k^{(a')}) \left(\sum_{i=1}^{N} p(\breve{\mathrm{x}}_k^{(a')}|\mathrm{x}_{k-1}^{(i)})\right)}$$

end

is a bidimensional case given by a linear state process with a nonlinear observation model giving rise to a banana-shaped likelihood function. The intent is to illustrate the applicability of the hybrid GH filter to nonlinear filtering problems for which particle filters are difficult to design.

1.8.1 One-dimensional example

Here we present an example widely used in the nonlinear filtering literature (e.g. [4]), described by the following set of equations

$$x_k = \frac{x_{k-1}}{2} + \frac{25x_{k-1}}{1+x_{k-1}^2} + 8\cos(1.2k) + u_k, \tag{1.34}$$

$$y_k = \frac{x_k^2}{20} + v_k, \tag{1.35}$$

where $\{u_k\}_{k\in\mathbb{N}_0}$ and $\{v_k\}_{k\in\mathbb{N}_0}$ are zero-mean Gaussian random processes with variances Q_k and R_k, respectively. To consider the same example as [4], we use $\mathrm{Q}_k = 10$ and $\mathrm{R}_k = 1$. Both the process simulation and the filtering are based on the same model, and the initialization of the filters assume that the state at $t = 0$ and the prior probability distribution are known. The hybrid Gauss-Hermite filter was designed with a simple procedure for scaling the grid, based solely on the range covered by the posterior set of samples. The optional accept-reject step was not used.

One exemplar run for the hybrid Gauss-Hermite filter is presented, illustrating the produced estimates. In addition, we quantify the performance of the hybrid Gauss-Hermite filter, in terms of the root mean squared error (RMSE), and compare it with those of a bootstrap (particle) filter and an UKF over 100 Monte Carlo runs. We set the bootstrap filter with 100 particles and the hybrid GH filter with 100 particles and 25 nodes. As noted in [4], although the RMSE is not meaningful for this multimodal problem, it indicates the accuracy of the filtering algorithms. The exemplar run is shown in Figure 1.1 and the comparison in terms of RMSE between the hybrid Gauss-Hermite filter, the bootstrap filter and the UKF is presented in Figure 1.2.

The run depicted in Figure 1.1 illustrates that the hybrid Gauss-Hermite filter provides accurate estimates to a highly nonlinear state process with a bimodal posterior distribution. The Figure 1.2 shows that the hybrid GH filter is as accurate as standard particle filters for this problem. Particle filters are known to provide outstanding results for this specific example [4], and so can be regarded as benchmark. It is important to note that performances of several variants of particle filters don't differ much for this problem [4] and, in general, all particle filters outperform the EKF and UKF, which legitimizes a comparison of the hybrid GH filter against a basic bootstrap particle filter. The RMSEs and runtimes averaged over all time steps, for all filters, are presented in Table 1.1. In this table, some accuracy gain can be perceived for the hybrid GH filter in terms of overall RMSE at the cost of a higher computational effort.

	RMSE	Runtime (s)
Bootstrap particle filter	3.1360	0.0022
UKF	9.3040	0.0001
Gauss-Hermite filter	2.7418	0.0189

Table 1.1: Average RMSE and runtime for the one-dimensional example

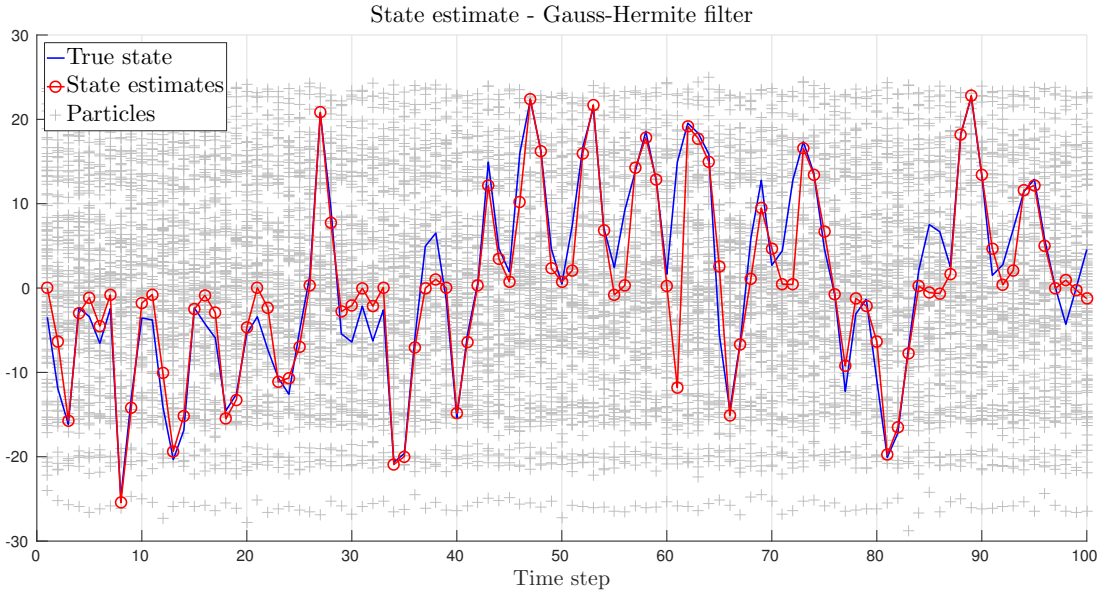


Figure 1.1: Exemplar run for the one-dimensional example

1.8.2 Two-dimensional example

This example illustrates the application of the hybrid GH filter for a multidimensional problem that was first proposed in [136]: tracking a single target by a single radar that provides accurate range measurements and fairly inaccurate angular measurements. In addition to simplicity, three aspects drove our choice for this example. The first is that it is sensible for the target tracking community, since it emulates a setting quite typical of real-world applications, that is, tracking with a radar that gives rise to the well-studied banana-shaped likelihood problem. The second aspect is that, as discussed in [136], when extrapolated to a multi-target scenario, the problem would require filters with a precise description of the posterior density in order to allow two targets closely spaced at high range to be discriminated. This could enable robust fusion of data from multiple spatially-separated and potentially mobile radars. The third reason is that the example is proposed with some exaggeration of the angular uncertainty (elevation or azimuth), posing a challenge for filters that approximate the posterior density by a Gaussian, e.g. EKF and UKF, and for standard particle filters.

The state is defined as $\mathrm{x}_k = (x_{1,k}, x_{2,k})^T$, where $(x_{1,k}, x_{2,k})$ are the Cartesian coordinates of the target at the time step $t = k\Delta t$. The state equation and the radar measurement equation are given by the following expressions, respectively

$$\mathrm{x}_k = \mathrm{x}_{k-1} + \mathrm{u}_k, \tag{1.36}$$

$$\mathrm{y}_k = \left(\sqrt{x_{1,k}^2 + x_{2,k}^2}, \arctan\left(\frac{x_{2,k}}{x_{1,k}}\right)\right)^T + \mathrm{v}_k. \tag{1.37}$$

where $\{\mathrm{u}_k\}_{k\in\mathbb{N}_0}$ and $\{\mathrm{v}_k\}_{k\in\mathbb{N}_0}$ are zero-mean Gaussian random processes with covariances respectively given by

$$\mathrm{Q}_k = \sigma_q^2 \begin{pmatrix} \Delta t & 0 \\ 0 & \Delta t \end{pmatrix}, \quad \mathrm{R}_k = \begin{pmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\theta^2 \end{pmatrix}, \tag{1.38}$$

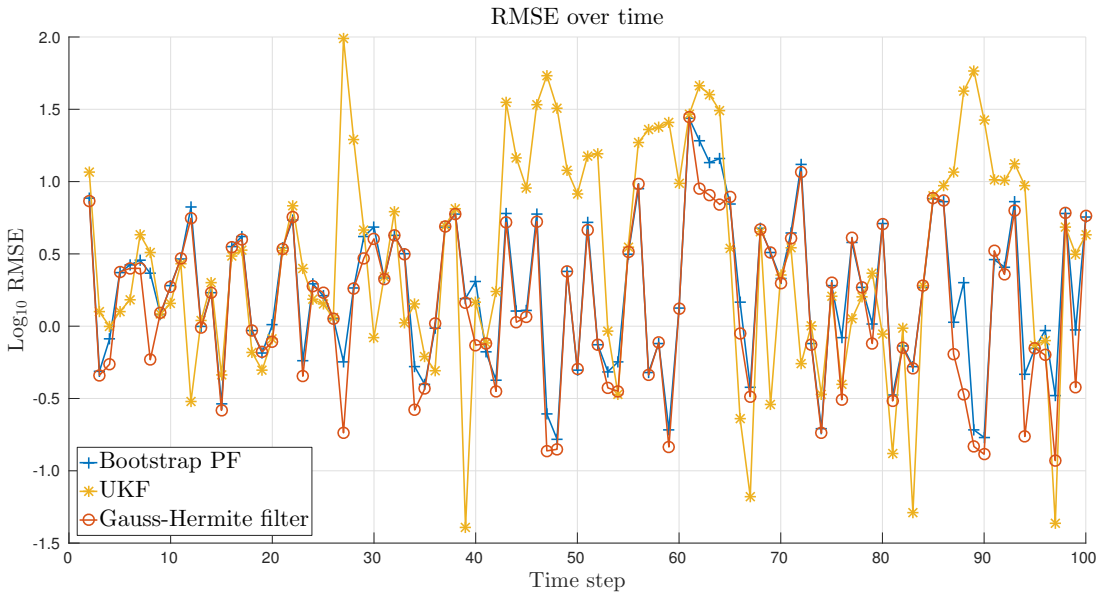


Figure 1.2: Comparison between different filters for the one-dimensional example

and where the amount of process noise is weighted by $\sigma_q^2 = 1\,\mathrm{m}^2/\mathrm{s}$ and the sampling period set as $\Delta t = 1\,\mathrm{s}$. The variances of range and elevation are respectively set as $\sigma_r^2 = 1 \times 10^{-3}\,\mathrm{m}^2$ and $\sigma_\theta^2 = 1\,\mathrm{rad}^2$. This model was used both for simulating the trajectory and for filtering. A single trajectory was simulated starting from $\mathrm{x}_0 = (100, 0)^T$.

Specifically for this example, in order to adequately propagate the particles for the hybrid GH filter, the samples are drawn in the sensor coordinate frame before approximating the posterior distribution. The same technique has been advocated in [136] to work around the scaling differences between range and elevation in the posterior density. We recall that the propagation of samples is required both for positioning the grid and generating the particles for the next step. In order to sample in the observation coordinate frame the filter uses a procedure akin to the Unscented Particle Filter (UPF). The UPF described in [195] instantiates one UKF per particle to construct a proposal density that targets the posterior density. Additionally, for the Gauss-Hermite filter, we chose to perform estimation in the radar coordinates, by setting the quadrature grid directly in the transformed space. This choice was based on the fact that, in Cartesian coordinates, it is very difficult to find an efficient grid configuration due to the exaggerated difference of accuracy between range and elevation. In the Cartesian state space, the shape of posterior density results in a scenario where either the grid is adequate for estimating range or elevation, but not both. Transforming to the radar coordinates is carried out by the unscented transform. The quadrature grid in the transformed state space and the corresponding grid expressed in the Cartesian space are illustrated in Figures 1.3a and 1.3b.

We compare the hybrid GH filter with the EKF, UKF, a particle filter with an UKF-based (sub)optimal proposal[6], and a particle filter with an UKF-based optimal proposal that generate samples directly in the radar coordinates. This latter is called hereafter *"transformed (sub)optimal proposal particle filter (UKF)"*, and corresponds to the filter developed in [136]. We consider this filter as a benchmark for this example.

All filters are initialized by setting the initial state estimate based on the first measurement,

[6]Suboptimal proposal in the sense of minimizing the weights' variance [64].

considered as normally distributed with the measurement noise covariance matrix converted to compose the state covariance. The initialization scheme is the same as that proposed in [136]: samples are drawn in the polar frame according to the measurement noise and converted to the Cartesian space. The particle filters are set with 250 particles, and the hybrid GH filter uses 250 particles and 50 nodes per dimension.

One exemplar run for the hybrid Gauss-Hermite filter is shown in Figure 1.4 to illustrate the produced estimates. The performance is quantified in terms of accuracy (root mean square error) for all evaluated filters over 200 Monte Carlo runs. A comparison in terms of RMSE between the hybrid Gauss-Hermite filter and all other filters is presented in Figure 1.5. The RMSEs and runtimes averaged over all time steps, for all filters, are presented in Table 1.2.

	RMSE (Cartesian)	RMSE (polar)	Runtime (s)
EKF	63.5547	0.7204	0.0001
UKF	53.9047	1.0959	0.0004
Opt. prop. PF (UKF)	86.6687	44.1717	0.0677
Transf. opt. prop. PF	12.8003	2.1737	0.0563
Gauss-Hermite filter	14.0669	0.1396	0.1853

Table 1.2: Average RMSE and runtime for the two-dimensional example

Regarding the exemplar run, the Figure 1.4 shows that the proposed filter provides accurate estimates for the radar problem, which is clearly difficult to achieve. Figure 1.5 shows that the hybrid GH filter presents similar performance, in terms of RMSE, as that of a particle filter that was especially designed for this example [136]. Though the hybrid GH filter presents some marginal gain in accuracy for estimates expressed in the polar frame, as consolidated in Table 1.2.

The results demonstrate that the hybrid GH filter is applicable to low-dimensional problems and can tackle problems generally difficult for standard particle filters and established nonlinear filters, such as the EKF and UKF. Note that this latter example is very difficult for any grid-based estimation method, and particularly difficult for the hybrid GH filter because of the sparse representation by only a few samples that induce the approximated posterior density. This fact explains the variations in the RMS errors over time since, due to the sparsity, some parts of the grid do not get enough information from samples. This suggests that increasing the number of particles could improve the overall performance.

One should consider that it is difficult to motivate the use of the hybrid GH filter for this degenerate sensor problem based on the root mean square error alone. As explained before, the merit of this example is rather illustrative and informative on how to apply the method, and is not intended to be a complete demonstration of the filter effectiveness. It is also worth noting that for both examples presented in this chapter, the Gauss-Hermite filter shows performance similar to that of particle filters, delivering some marginal gain in accuracy. This seems intuitive because the estimation errors are dominated by errors in the sequential Monte Carlo representation of the target density. Most likely this is due to the inherent sparsity of the probability masses, whose resulting error is likely to be transferred to the grid description.

1.9 Conclusions

This chapter was intended to address the effective representation of probability distributions for highly nonlinear and non-Gaussian processes, as an important challenge identified in Chapter 0. Specifically, this has been achieved by a novel estimation algorithm that has been derived. The algorithm is based on a different idea: a hybrid method that combines a particle-based representation of the prior state uncertainty with an efficient grid-based method to estimate the posterior probability density. The method makes use of the prior Monte Carlo empirical measure to induce a probability mass function that approximates the posterior probability measure. This probability mass function enables accurate numerical integration, by means of the Gauss-Hermite quadrature, to compute the state estimate and covariance matrix.

As touched on before, the estimation method is not based on the importance sampling principle and so the hybrid Gauss-Hermite filter avoids the *weight degeneracy* problem, which is inherent to particle filters. Additionally, the estimation problem is solved without resorting to techniques for distribution matching, classical procedures of grid centering and refinement, and without parameterization of target densities as Gaussians.

The filter was shown to be suitable to tackle the estimation problem for nonlinear and/or non-Gaussian processes and measurement models, including cases when the posterior densities are not unimodal. Some factors need to be taken into account when considering the hybrid GH filter. First, it requires a study on how accuracy could be traded with complexity to deem the filter feasible for practical use. The second factor, which is also applicable for designing particle filters, involves filtering settings that must be analyzed on a case-by-case basis to determine how the problem dimensionality degrades estimation performance. Nevertheless, in general, should the filter be understood as feasible, the computational effort should not be an obstruction if because the algorithm is highly parallelizable. A number of techniques for parallel computing have become very accessible recently, including cloud and multi-core computing, and general-purpose graphics processing units (GPGPU).

References

[4] M. S. Arulampalam et al. “A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking”. In: *IEEE Transactions on Signal Processing* 50.2 (2002), pp. 174–188.

[11] N. Bergman. “Recursive Bayesian Estimation: Navigation and Tracking Applications”. PhD thesis. Linköping University, Department of Electrical Engineering, 1999.

[12] A. Beskos et al. “Error Bounds and Normalising Constants for Sequential Monte Carlo Samplers in High Dimensions”. In: *Advances in Applied Probability* 46.1 (2014), pp. 279–306.

[21] R. S. Bucy and K. D. Senne. “Digital Synthesis of Non-linear Filters”. In: *Automatica* 7.3 (May 1971), pp. 287–298.

[26] O. Cappé, S. J. Godsill, and É. Moulines. “An Overview of Existing Methods and Recent Advances in Sequential Monte Carlo”. In: *Proceedings of the IEEE* 95.5 (May 2007), pp. 899–924.

[28] S. Challa, Y. Bar-Shalom, and V. Krishnamurthy. "Nonlinear filtering via generalized Edgeworth series and Gauss-Hermite quadrature". In: *IEEE Transactions on Signal Processing* 48.6 (2000), pp. 1816–1820.

[34] J. Cornebise, É. Moulines, and J. Olsson. "Adaptive methods for sequential importance sampling with application to state space models". In: *Statistics and Computing* 18.4 (Dec. 2008), pp. 461–480.

[46] F. Daum and J. Huang. "Curse of dimensionality and particle filters". In: *IEEE Aerospace Conference, 2003*. Vol. 4. Mar. 2003, pp. 1979–1993.

[47] F. Daum and J. Huang. "Particle flow for nonlinear filters, Bayesian decisions and transport". In: *FUSION 2013, Proceedings of the 16th International Conference on Information Fusion*. 2013, pp. 1072–1079.

[49] F. Daum and J. Huang. "Nonlinear filters with log-homotopy". In: *Proc. SPIE 6699, Signal and Data Processing of Small Targets 2007*. Vol. 6699. Signal and Data Processing of Small Targets, 2007. San Diego, California, Sept. 2007, pp. 669918–669918-15.

[55] P. Del Moral. *Feynman-Kac Formulae: Genealogical and Interacting Particle Systems with Applications*. New York: Springer, 2004.

[64] A. Doucet, S. Godsill, and C. Andrieu. "On sequential Monte Carlo sampling methods for Bayesian filtering". In: *Statistics and Computing* 10.3 (July 2000), pp. 197–208.

[66] A. Doucet and A. M. Johansen. "A Tutorial on Particle Filtering and Smoothing: Fifteen Years Later". In: Oxford Handbook of Nonlinear Filtering. 2011, pp. 656–704.

[83] N. J. Gordon, D. J. Salmond, and A. F. M. Smith. "Novel approach to nonlinear/non-Gaussian Bayesian state estimation". In: *IEEE Proceedings - Radar and Signal Processing* 140.2 (Apr. 1993), pp. 107–113.

[97] K. Ito and K. Xiong. "Gaussian filters for nonlinear filtering problems". In: *IEEE Transactions on Automatic Control* 45.5 (May 2000), pp. 910–927.

[98] K. Ito and K. Xiong. "Gaussian Filters for Nonlinear Filtering Problems". In: *IEEE Transactions on Automatic Control* 45 (2000), pp. 910–927.

[99] P. Jäeckel. *A Note on Multivariate Gauss-Hermite Quadrature*. May 2005. URL: http://jaeckel.16mb.com/ANoteOnMultivariateGaussHermiteQuadrature.pdf.

[102] B. Jia, M. Xin, and Y. Cheng. "Sparse-grid Quadrature Nonlinear Filtering". In: *Automatica* 48.2 (Feb. 2012), pp. 327–341.

[107] S. J. Julier and J. K. Uhlmann. "Unscented Filtering and Nonlinear Estimation". In: *Proceedings of the IEEE*. 2004, pp. 401–422.

[110] M. Klaas, N. de Freitas, and A. Doucet. "Toward Practical N^2 Monte Carlo: the Marginal Particle Filter". In: *ArXiv e-prints* (July 2012). arXiv: 1207.1396 [stat.CO].

[111] J. H. Kotecha and P. M. Djurić. "Gaussian particle filtering". In: *IEEE Transactions on Signal Processing* 51 (2003), pp. 2592–2601.

[112] J. H. Kotecha and P. M. Djurić. "Gaussian sum particle filtering". In: *IEEE Transactions on Signal Processing* 51.10 (2003), pp. 2602–2612.

[115] H. J. Kushner and A. S. Budhiraja. "A Nonlinear Filtering Algorithm Based on an Approximation of the Conditional Distribution". In: *IEEE Transactions on Automatic Control* 45 (1999), pp. 580–585.

[118] F. Le Gland and N. Oudjane. "A Sequential Particle Algorithm that Keeps the Particle System Alive". In: Stochastic Hybrid Systems: Theory and Safety Critical Applications. Springer Berlin Heidelberg, 2006, pp. 351–389.

[124] Q. Liu and D. A. Pierce. "A note on Gauss-Hermite quadrature". In: *Biometrika* 81.3 (1994), pp. 624–629.

[136] S. Maskell et al. "Tracking using a radar and a problem specific proposal distribution in a particle filter". In: *IEEE Proceedings - Radar, Sonar and Navigation*. Vol. 152. Institute of Electrical and Electronics Engineers Inc., Jan. 2005. Chap. 5, pp. 315–322.

[145] C. Musso, N. Oudjane, and F. Le Gland. "Improving Regularised Particle Filters". In: *Sequential Monte Carlo Methods in Practice*. Ed. by A. Doucet, N. de Freitas, and N. Gordon. Sequential Monte Carlo Methods in Practice. New York: Springer-Verlag, 2001. Chap. 12, pp. 247–271.

[146] R. M. Neal. "Annealed Importance Sampling". In: *ArXiv e-prints* (Mar. 1998).

[156] M. K. Pitt and N. Shephard. "Filtering via Simulation: Auxiliary Particle Filters". In: *Journal of the American Statistical Association* 94.446 (1999), pp. 590–599.

[157] W. H. Press et al. *Numerical Recipes in C: The Art of Scientific Computing*. New York, NY, USA: Cambridge University Press, 1988.

[161] S. Reich. "A dynamical systems framework for intermittent data assimilation". English. In: *BIT Numerical Mathematics* 51.1 (2011), pp. 235–249.

[166] B. Ristic, S. Arulampalam, and N. Gordon. *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Artech House, 2004.

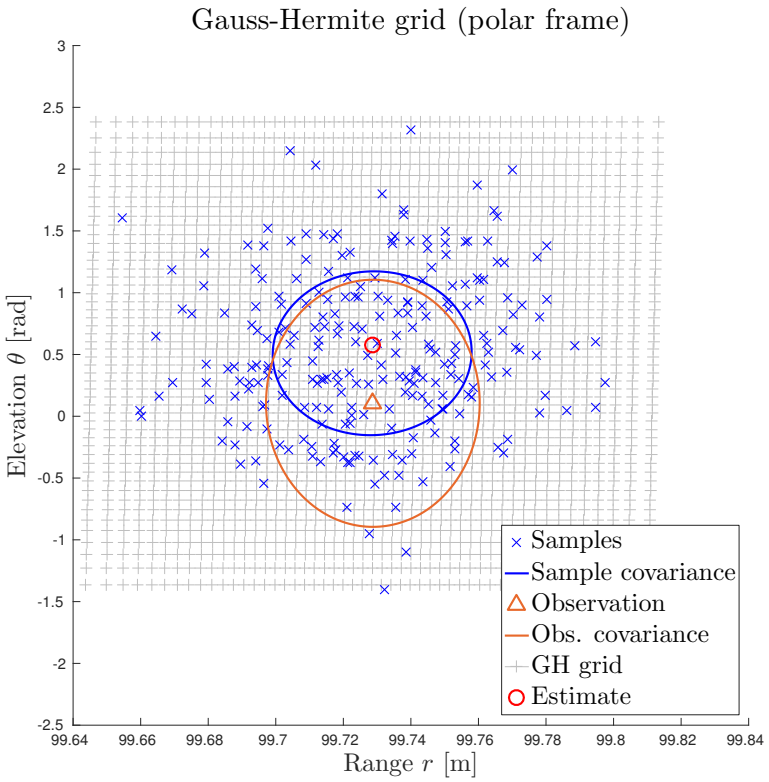
[171] H. P. Saal, N. M. O. Heess, and S. Vijayakumar. "Multimodal Nonlinear Filtering Using Gauss-Hermite Quadrature". In: *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2011, Athens, Greece, September 5-9, 2011, Proceedings, Part III*. Ed. by D. Gunopulos et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 81–96.

[178] M. Šimandl, J. Královec, and T. Söderström. "Advanced point-mass method for nonlinear state estimation". In: *Automatica* 42.7 (July 2006), pp. 1133–1145.

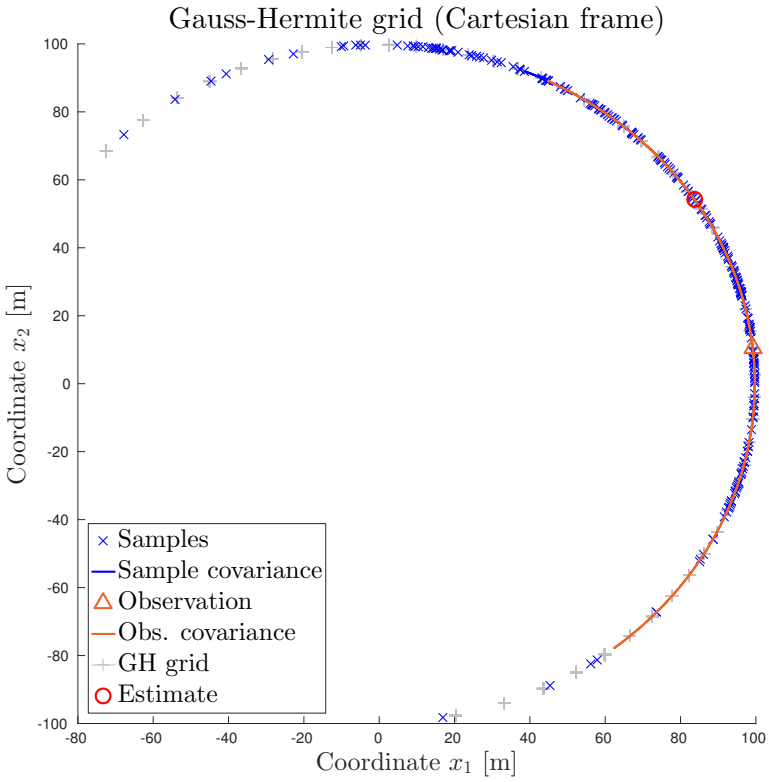
[189] A. H. Stroud and D. Secrest. *Gaussian quadrature formulas*. Englewood Cliffs, NJ: Prentice-Hall, 1966.

[195] R. van der Merwe et al. *The Unscented Particle Filter*. Tech. rep. CUED/F-INFENG/TR 380. Cambridge University, Engineering Department, 2000.

[210] C. Wu and C. Han. "Quadrature Kalman Particle Filter". In: *Journal of Systems Engineering and Electronics* 21.2 (2010), pp. 175–179.

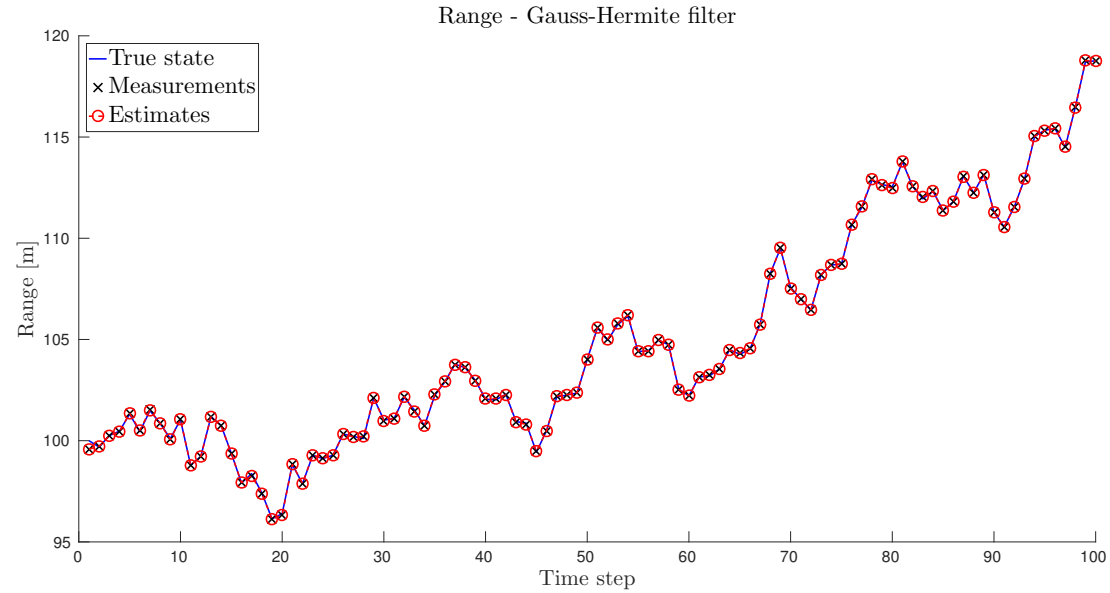


(a) Gauss-Hermite grid in polar coordinates

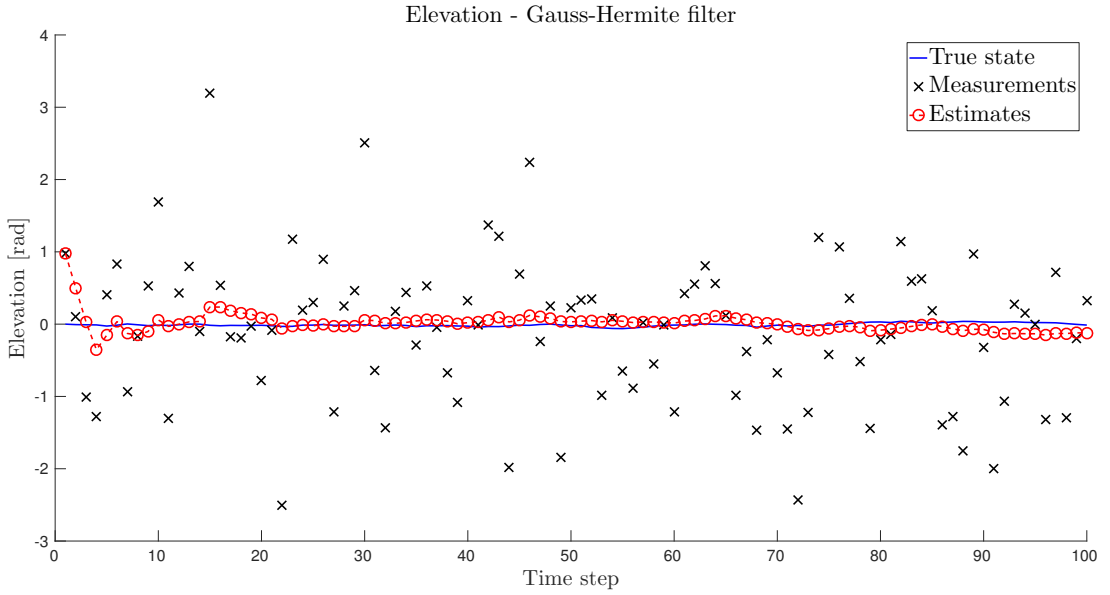


(b) Gauss-Hermite grid in Cartesian coordinates

Figure 1.3: Gauss-Hermite grid

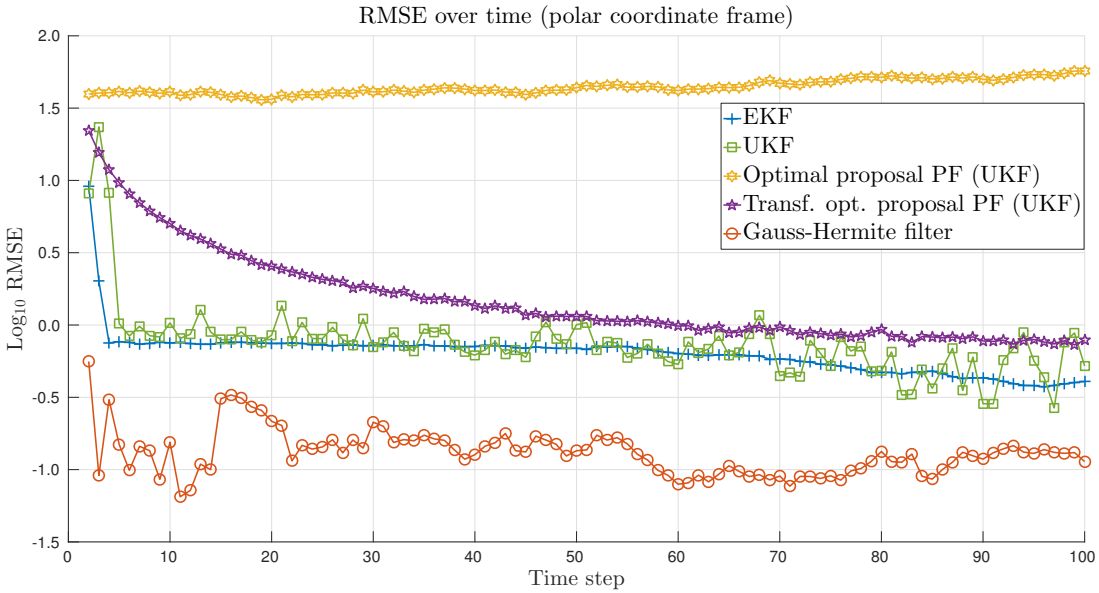


(a) Range estimates

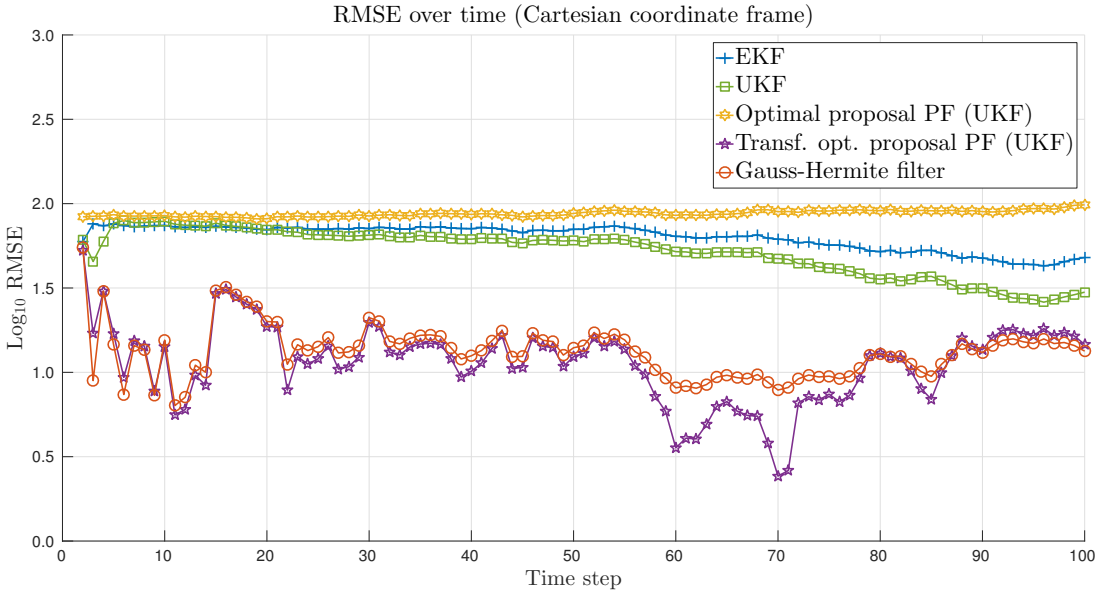


(b) Elevation estimates

Figure 1.4: Exemplar run for the two-dimensional example



(a) RMSE in polar coordinates



(b) RMSE in Cartesian coordinates

Figure 1.5: Accuracy of all filters for the two-dimensional example

2

Stochastic Particle Flow

Among the challenges presented in Chapter 0 for multi-object estimation, one, in particular, is of crucial importance for real-world applications: that of describing the filtering probability distributions of high-dimensional nonlinear phenomena while mitigating part of the inaccuracies that arise from the curse of dimensionality, and irregularities caused by usual Monte Carlo approximations. To address this challenge, this chapter proposes a fundamental technique that envisages tackling high-dimensional, highly nonlinear problems, and coping well with scenarios of multimodality, high state uncertainty, uninformative observations, and low signal-to-noise ratio.

The context for introducing the technique is established by a series of novel filters for the probabilistic inference that proposes an alternative way of performing Bayesian updates, called particle flow filters. These filters provide approximate solutions to nonlinear filtering problems. They do so by defining a continuum of densities between the prior probability density and the posterior, i.e. the filtering density. Building on these methods' successes, we propose a novel filter. This new filter aims to address the shortcomings of sequential Monte Carlo methods when applied to important nonlinear high-dimensional filtering problems. The novel filter uses equally weighted samples, each of which is associated with a local solution of the Fökker-Planck equation. This hybrid of Monte Carlo and local parametric approximation gives rise to a global approximation of the filtering density of interest. In this chapter, we show that, when compared with state-of-the-art methods, the Gaussian-mixture implementation of the new filtering technique, which we call Stochastic Particle Flow, has utility in the context of benchmark nonlinear high-dimensional filtering problems.

2.1 Overview

Stochastic filtering in high-dimensional spaces is a challenging estimation task because of two fundamental issues:

- *The curse of dimensionality.* In a statistical experiment, as the sample space's dimensionality increases a finite number of realizations can only populate the space to an increasingly sparse extent [149]. This issue makes it challenging to use approximations based on realizations of the state.

- *The infinite number of parameters required to describe a general probability density on a continuous state space.* Such a density, in common with any other real function, can always be exactly described using a power series with infinitely many terms. In all but a few cases, where the density is known to have a specific parametric form, using a finite set of parameters is necessarily an approximation to this complete description. The fidelity of

such approximations falls rapidly as dimension increases. This issue makes it challenging to define a parameterization that uses a number of parameters that scales only gently with dimension.

The development of the vast majority of practical filters focuses on how to accurately represent generic probability densities. However, in the view of the author, relatively few filters are systematically developed with the explicit intent of efficiently expressing densities in high-dimensional spaces. There does appear to be a consensus that the statistical efficiency associated with expressing high-dimensional filtering densities can be improved by simulating tempering distributions [79, 62, 147, 56]. Such approaches involve introducing intermediate distributions such that it is easier to migrate between these intermediate distributions than it is to migrate directly from the prior to the posterior. The use of such intermediate distributions stabilizes the sampling procedure and maintain the variance of the Monte Carlo weights at an acceptable level. Bickel *et al.* [14] considered, in the context of a bootstrap particle filter, the number of intermediate distributions needed to use such tempering successfully. They prove that, as the dimensionality increases, the number of intermediate distributions needed to accurately represent a high-dimensional density becomes practically infinite. This implies that considering a continuum, i.e. infinite number, of intermediating distributions [147] might be the basis of a successful approach. This implication is corroborated by the reported success of Markov chain Monte Carlo (MCMC) algorithms that populate high-dimensional state spaces efficiently using approximations of problem-specific continuous-time processes [67, 148, 168, 80][1].

Techniques for continuous-time processes stem from the seminal work by Stratonovich [187], Kushner [114] and Zakai [211] on filtering theory. The most popular instance of such filters is the so-called Kalman-Bucy filter [20], the continuous-time counterpart of the Kalman filter. More general filters directly approximate solutions to the Kushner-Stratonovich equation either by a finite-dimensional density parameterization [10, 54] or by Monte Carlo methods [38, 36, 39]. Other important finite-order filters that appeal to an unusual formalism of multiple stochastic integrals [22, 23] are worth mentioning as well.

Continuous-time filtering may be seen by some as an idealized problem of limited practical utility. However, recent research [198, 160] has shown that continuous-time filtering can offer key insights into the fundamental principles necessarily associated with successful filtering in high-dimensions: the effects of local, continuous spatial properties of the observation process need to be incorporated in the solution. As identified by Bickel *et al.* [14], the information in the data gives rise to the notion of the *effective dimension* of the space. It is this effective dimension, and not the dimension of the state space itself, that actually affects the statistical efficiency of inference algorithms. This implies that tempering only addresses part of the problem, remaining the local observation properties to be incorporated. Recently, based on a principled approach, Rebeschini & van Handel [160] proposed to decompose the state space into separate blocks. The global solution to the inference problem is then constructed by combining the local solutions for each of the blocks. The chapter goes on to demonstrate that, by using the *decay of correlations* property[2], it is possible to develop particle filters based on local solutions in such a way that

[1] Note that while practical implementation of these techniques necessarily involves finite time-horizons, the continuous-time processes are typically designed such that, as time tends to infinity, the distribution of the samples from the process tends to the distribution of interest. This is in contrast to the use of tempering distributions, where samples from the posterior are generated after a defined (finite) number of steps between intermediate distributions.

[2] A spatial counterpart of the stability property of nonlinear filters, by which a probability mass is strongly

the approximation error does not depend on the dimension of the state space.

Largely independently, the idea of filtering via a continuum of intermediate distributions seems to have its appeal reinvigorated as several new methods have been proposed for progressive Bayesian updates, whose continuity is considered in the limit, aiming to gradually introduce the effect of each observation. Those filters have emerged either in a variational, ensemble-based, or sequential Monte Carlo framework. In the variational framework, the new methods presented in [89, 87, 88] pose the filtering problem as a multi-step optimization problem for which the cost function is an approximated distance between a parameterized density and the actual filtering density. In the ensemble-based framework, the methods [161, 162, 197] are focused on data assimilation problems and apply ideas of optimal transport along with continuous-time filtering to generate multiple independent solutions that are combined to obtain a single solution of an inference problem. The methods in the sequential Monte Carlo framework explore extensions or alternatives to particle filters (e.g., [151, 35, 49, 172, 37]), or simply capitalize on techniques for properly choosing a sequence of *bridging* importance densities (e.g., [62, 147, 56, 24]), carrying on the intent to overcome the widely known problem of particle filters called *degeneracy* or collapse of weights [4, 46, 183, 65].

Among those methods, one, in particular, has recently attracted interest in an Engineering context where it has been described as *particle flow*. The performance that has been reported is remarkable and the literature is extensive with several variants having been developed over recent years (see, for example, [49, 51, 50, 48, 47]). The development of particle flow draws on analogies to problems that arise in Fluid Dynamics and Electromagnetism. These filters *flow* probability masses (particles) from a prior probability space to one that is updated according to a set of measurements without the need to perform a Bayesian update explicitly. All particle flow algorithms explore the concept of a homotopy between the prior and posterior probability spaces, implicitly describing a joint measure that couples the prior and posterior probability measures. This idea is in the heart of the Kantorovich's optimal transportation problem [199] that, by evoking deterministic transport maps for very simple cost functions and dynamic constraints, yields an essential explanation on why original particle flow methods work well based on deterministic rules to flow the particles.

When the sequential filtering problem involves non-compactly supported densities, solving it via deterministic (optimal) transport is not straightforward. A solution would require either a non-trivial approximation of the highly nonlinear Monge-Ampère equation [86] or adapting classical solutions constructed for measures on bounded sets [142, 40]. In these approaches, severe technical difficulties may arise and not all the effects on the estimation errors are clearly known. A continuously evolving, exact, optimal transport map would require a complete description, at all time instants, of an embedding dynamic field that induces a transference plan to correctly move particles. If the posterior density could be completely characterized beforehand then the optimal transport problem could be numerically solved by the multiple-step augmented-Lagrangian optimization method as proposed by Benamou & Brenier [9]. However, detailed knowledge of the posterior would imply a direct answer to the filtering problem. As an alternative, theoretically speaking, a complete description of the optimal field could be achieved by solving the Monge-Ampère equation for any possible location of particles on the state space. Notwithstanding, the Monge-Ampère equation admits exact solutions only for few particular

correlated to masses within its neighborhood but has negligible correlation with respect to the remaining areas of the state space.

cases [86] and would also require a thorough description of the posterior density in advance.

In this scenario, one feasible approach is that advocated by particle flow methods, which take simplifying assumptions on the embedding dynamic field in order to avoid both an optimization over a parametric class of transport maps and an explicit solution of the associated elliptic partial differential equation. However, in our experience, these simplifying assumptions result in approximated filtering densities providing accurate estimates for the first-order moment but inconsistent estimates for second and higher-order moments, whose quality is highly dependent on the problem and algorithm settings (e.g., [91]). In practice, particle flow methods address this latter issue by either relying on a companion filter [30, 63] or using the sample covariance matrix with shrinkage and Tikhonov regularization [109] to be able to estimate the second-order moment.

We conjectured if appealing to stochastic transport could provide a new avenue for solving the filtering problem. Fortunately, a variational formulation of the Fökker-Planck equation as a gradient flow, as exposed by Jordan *et al.* [106], enables the precise interpretation that, if a transport operation is to be understood as a diffusion, then it minimizes the free energy functional of the process with respect to the Wasserstein metric over an admissible class of probability measures. Relying on this formulation, it is straightforward to obtain a transport rule, optimal in terms of minimizing the free energy functional, as a Langevin stochastic process. This rule is based simply on the assumptions of stationarity of the filtering distribution (Gibb's distribution) and on potential conditions, for which an embedding stationary field is exactly derived.

In this chapter, we take into consideration the findings presented by Jordan *et al.* [106], incorporate the description of statistically efficient processes in high-dimensional spaces as proposed by Girolami & Calderhead [80], and incorporate local properties of the observation process to formulate a stochastic variant of particle flow[3]. This new *stochastic particle flow* (SPF) involves defining a Langevin diffusion such that a posterior measure from a previous step, under a known stationary potential field, is diffused onto the current posterior measure, satisfying the Fökker-Planck equation to produce an accurate approximation of the filtered density. This process involves guiding local solutions of the Fökker-Planck equation in such a way that we construct a mixture that approximates the posterior. As we will discuss later on, the SPF method we propose is essentially built as a Gaussian sum filter (SPF-GS), nevertheless, it is possible to use a similar formulation to define an implementation strategy based on a marginal particle filter (SPF-MPF). This variant demonstrates the versatility of the SPF to algorithm settings.

It is worth mentioning that our resulting SPF technique is in the same ethos as the method recently developed by Bunch & Godsill [24, 25]. However, in contrast to our approach, their method (i) is based on the homotopy between the prior and posterior spaces, (ii) assumes the particle flow is an Ornstein-Uhlenbeck process whose scaling parameter determines the rate of diffusion of samples' paths, (iii) proposes weights that must be updated iteratively by a partial differential equation (PDE) describing how the unnormalized log-density evolves with a pseudo-time variable; (iv) is articulated as a standard (not marginal) particle filter.

The outline of the chapter is as follows. We begin by reviewing the stochastic filtering problem in a sequential Monte Carlo framework in Section 2.2. We abstract the solution in

[3]Existing particle flow algorithms (including, perhaps surprisingly, that known as non-zero diffusion particle flow [52]) propagate particles deterministically.

terms of a general map that could adopt any valid method to perform the filtering update. In Section 2.3, we present a brief overview of the original particle flow methods. We discuss their principles in order to further clarify these methods and motivate the natural step towards the stochastic particle flow. In Section 2.4 we derive the generic SPF algorithm by describing the proposed dynamics of probability masses, describing the associated stationary solution to the Fökker-Planck equation, and constructing the stochastic flow. Algorithmic details are given and relate to how to compute the diffusion matrix, how to integrate the stochastic flow, and how to select the simulation time horizon and integration step size. We present the stochastic particle flow implementation using a Gaussian sum filter (SPF-GS) in Section 2.5. We achieve this by considering the posterior to be well approximated as a mixture of local solutions to the flow. Similarly, in Section 2.6 we show the SPF articulated as a marginal particle filter (SPF-MPF) by setting the importance density as a mixture analogous to that generated by the SPF-GS. Section 2.7 then illustrates the SPF's properties by a series of toy problems, and compares the performance of the SPF and other state-of-the-art methods in the context of three instructive multi-sensor or multi-target tracking problems: multi-sensor bearing-only tracking, convoy tracking and inference on a large network of sensors (as in [175]). In the comparisons for the multi-sensor bearing-only and convoy tracking problems, we included extensions to two of the most effective (original) particle flows, namely, the Gaussian particle flow (GPF) [53] and the scaled-drift particle flow (SDPF) [52]. Finally, Section 2.8 concludes.

2.2 Sequential Monte Carlo Filtering

In this section, we report the filtering framework within which the particle flows may be formalized. Let $\{\mathrm{x}_t \in \mathcal{X} : t \in \mathbb{R}_+\}$ be a sequence of states generated through time by a known continuous-time state process, modeled as a Markov process, and let $\{\mathrm{y}_{t_k} \in \mathcal{Y} : t_k \in \mathbb{R}_+, k \in \mathbb{N}\}$ be a sequence of discrete-time observations of the process generated by an observation model. In the classical filtering problem, one is required to compute the best estimate of a function of interest φ of the state, given all observations realized up to the time instant t_k, i.e.,

$$\hat{\varphi}_k = \mathbb{E}\left[\varphi(\mathrm{x}_{t_k})|\mathrm{y}_{t_1}, \mathrm{y}_{t_2}, \ldots, \mathrm{y}_{t_k}\right]. \tag{2.1}$$

Now consider a set of particles $\{\mathrm{x}_{k-1}^{(i)}, w_{k-1}^{(i)}\}_{i\in[1..N]}$ constituting samples that can be used to approximate a filtering probability density $p(\mathrm{x}_{k-1}|\mathrm{y}_{1:k-1})$ by means of a Monte Carlo measure satisfying

$$\sum_{i=1}^{N} w_{k-1}^{(i)} \delta(\mathrm{x}_{k-1} - \mathrm{x}_{k-1}^{(i)}) \overset{N\to\infty}{\longrightarrow} p(\mathrm{x}_{k-1}|\mathrm{y}_{1:k-1}). \tag{2.2}$$

Given a new observation obtained at instant k, one wishes to find a procedure to transform the set of particles $\{\mathrm{x}_{k-1}^{(i)}, w_{k-1}^{(i)}\}_{i\in[1..N]}$ into a new set of particles $\{\mathrm{x}_k^{(i)}, w_k^{(i)}\}_{i\in[1..N]}$ that incorporates the effect of the latest observation to estimate the filtered entity as

$$\hat{\varphi}_k \approx \sum_{i=1}^{N} w_k^{(i)} \varphi(\mathrm{x}_k^{(i)}). \tag{2.3}$$

In theory, the filtering problem in the sequential Monte Carlo form can be solved by any

map $\mathcal{T} : \mathcal{X} \times \mathcal{Y} \to \mathcal{X}'$, $\mathcal{T} \in \mathcal{C}^1(\mathbb{R}^{d_x}) \times \mathcal{C}^0(\mathbb{R}^{d_y})$, where $|\mathcal{X}'| = |\mathcal{X}|$, that implements

$$\begin{aligned} \mathrm{x}_k^{(i)} &:= \mathcal{T}(\mathrm{x}_{k-1}^{(i)}, \mathrm{y}_k);\ i = 1, \ldots, N; && (2.4) \\ w_k^{(i)} &:= \det \mathcal{J}_{\mathrm{x}_{k-1}} [\mathcal{T}]^{-1} w_{k-1}^{(i)}; && (2.5) \end{aligned}$$

where $\mathcal{J}_{\mathrm{x}_{k-1}}[.]$ is the Jacobian matrix with respect to x_{k-1}, and such that

$$\sum_{i=1}^{N} w_k^{(i)} \delta(\mathrm{x}_k - \mathrm{x}_k^{(i)}) \overset{N\to\infty}{\longrightarrow} p(\mathrm{x}_k|\mathrm{y}_{1:k}). \tag{2.6}$$

Although most practical filters implement the mapping (2.4) in terms of discrete Bayesian updates, there should be no objection to the general idea of considering the map $\mathcal{T}$ as a transform continuous in time within $t_{k-1} < t \leq t_k$. This idea establishes the basis for the particle flow filters.

2.3 Particle Flow

This section aims to present a brief overview of the particle flow methods, to discuss their principles, and to set the background for the introduction of the stochastic particle flow. The key idea of the particle flow is to transfer a set of probability masses by an operation that transports the prior probability measure onto the posterior measure. This operation realizes the measurement update smoothly in order to express a filtering entity, usually an estimate. The mechanism implied is, therefore, a filtering algorithm that avoids the need to perform a Bayesian measurement update explicitly.

Given a set of particles $\{\mathrm{x}^{(i)}(\lambda) \in \mathbb{R}^{d_x}\}_{i\in[1..N]}$ dependent on a continuous pseudo-time variable $\lambda \in [0, 1]$, where d_x is the number of dimensions of the state space, and such that $\mathrm{x}^{(i)}(0) = \mathrm{x}_{k-1}^{(i)}$ and $\mathrm{x}^{(i)}(1) = \mathrm{x}_k^{(i)}$, the transformation of the particles is accomplished by solving through $0 < \lambda \leq 1$ an ordinary differential equation (ODE) referred to as the flow equation

$$\frac{d\mathrm{x}}{d\lambda} = \mu(\mathrm{x}, \lambda), \quad \mathrm{x}^{(i)}(0) \sim p_0(\mathrm{x}). \tag{2.7}$$

The varieties of particle flow methods rely on how one defines the flow drift $\mu(\mathrm{x}, \lambda)$, which in turn depends on the assumptions made to solve the associated continuity equation

$$\frac{\partial p}{\partial \lambda} = -\nabla_{\mathrm{x}} \cdot (\mu \cdot p), \quad p(\mathrm{x}, 0) = p_0(\mathrm{x}). \tag{2.8}$$

The operator $\nabla_{\mathrm{x}} \cdot (.)$ is the divergence operator and the drift can be understood as a vector field $\mu(\mathrm{x}, \lambda) \in \mathbb{R}^{d_x}$ that is not uniquely determined for a given probability density $p(\mathrm{x}, \lambda)$. In the optimal transportation literature, the vector field is usually determined by the constraint that it minimizes the kinetic energy. In that case, the flow equation (2.7) can be written in terms of a dynamic potential field as $\mu(\mathrm{x}, \lambda) = \mathcal{M}^{-1}\nabla_{\mathrm{x}}\psi(\mathrm{x}, \lambda)$ [199], where $\mathcal{M}$ is a positive-definite mass matrix, ∇_{x} is the gradient operator, and $\psi(\mathrm{x}, \lambda)$ is a dynamic potential function that satisfies the elliptic PDE

$$\nabla_{\mathrm{x}} \cdot \left(p(\mathrm{x}, \lambda)\, \mathcal{M}^{-1}\nabla_{\mathrm{x}}\psi(\mathrm{x}, \lambda)\right) = -\partial_\lambda p(\mathrm{x}, \lambda). \tag{2.9}$$

An exact solution to equation (2.9) has been derived by Reich [161] considering Gaussian likelihood functions. In more general settings, if the target posterior density $\pi(\mathrm{x})$ could be thoroughly characterized in advance, the numerical solution to this problem could be achieved by the multiple-step augmented-Lagrangian optimization method as proposed by Benamou & Brenier [9]. However, availability of a detailed description of the posterior density would constitute a direct answer to the filtering problem. Similarly, the well-known flow constructed by Dacorogna & Moser [40], appropriate for mapping measures on bounded open sets, could be adapted for problems involving non-compactly supported densities as the solution of the p-Laplacian equation [70]

$$\nabla_{\mathrm{x}} \cdot (a(\mathrm{x}, \lambda) \nabla_{\mathrm{x}} \vartheta(\mathrm{x}, \lambda)) = \pi(\mathrm{x}) - p(\mathrm{x}, \lambda), \tag{2.10}$$

where $p(\mathrm{x}, \lambda)$ and $\pi(\mathrm{x})$ are the intermediate and target densities respectively. Function $a(\mathrm{x}, \lambda) \geq 0$, $a(\mathrm{x}, \lambda) \in \mathbb{L}^{\infty}$ ($\mathbb{L}^{\infty}$-space[4]), is a Lagrange multiplier that scales the distance of optimal transportation, whereas the term $\nabla_{\mathrm{x}} \vartheta(\mathrm{x}, \lambda)$ gives the direction of optimal transportation. As mentioned before, these transport-based solutions are not straightforwardly applicable to filtering problems as they would require anticipating the target probability density, and the solution by Dacorogna & Moser [40] would require truncation of the involved densities to bound their support.

Indeed, original particle flows do not follow the classical transport-based methodology but rather take simplifying assumptions on the dynamic potential field, avoiding the complexity of solving the elliptic PDEs (2.9) and (2.10). Specifically, the particle flows are derived from a programmed sequence of a dynamic potential field that roughly solves the equation (2.8). As examples we refer the reader to the incompressible particle flow [49], the Gaussian or exact particle flow [53], and the non-zero *"diffusion"* particle flow [52], which is not actually a diffusion, but simply takes into account a diffusion term to scale and/or offset the drift term.

In a closely related problem, as an alternative to the solution of elliptical equations or to original particle flows, it is possible to demonstrate that if the drift solves the continuity equation (2.8), under a stationary potential field (conservative) related to an invariant, locally[5] log-concave density of the form $p(\mathrm{x}, T) = \pi(\mathrm{x}) \propto \exp(-\psi(\mathrm{x}))$, then the flow (2.7) produces the maximum-a-posteriori (MAP) estimate, $\hat{\mathrm{x}}_{MAP}$, after an appropriate time horizon $\lambda \geq T$ (see *Theorem 2.12* in Section 2.9.2). A similar concept is used in optimization algorithms based on gradient descent. An evident problem with this approach is that, regardless of providing a MAP estimate, it is unable to capture higher-order aspects of a target posterior density. Thus, under the assumption of a stationary potential field, a stochastic particle flow seems suitable to describe a filtering density precisely up to an arbitrary moment order, by following the dynamics of a diffusion that minimizes the free energy functional (see [106] for details). Such stochastic flow would propagate a probability density according to the Fökker-Planck equation. This observation becomes fundamental when we note that, loosely speaking, obtaining a precise approximation of a stationary potential field requires less effort than obtaining a sequence of accurate approximations of a dynamic potential field. In this context, approximating a dynamic potential field forms the basis for the classical transport methodology (e.g., [161]).

[4]The $\mathbb{L}^{\infty}$-space generalizes the $\mathbb{L}^{p}$-spaces to $p = \infty$. An $\mathbb{L}^{p}$-space describes the set of all functions f for which the norm $\|f\|_{p} = \left(\int_{\mathcal{X}} |f|^{p}\right)^{1/p}$ converges. The concept is analogous for the $\mathbb{L}^{\infty}$-space although its norm is defined by the essential supremum.

[5]Log-concave in the vicinity of the density maxima.

2.4 Stochastic Particle Flow

This section derives the stochastic particle flow based on a stationary solution to the Fokker Planck equation. We capitalize on the fact that, under certain conditions on the drift and diffusion terms of a stochastic process, there is a stationary solution that satisfies a variational principle, minimizing a certain convex free energy functional over an admissible class of probability densities. The Fokker–Planck equation is shown to follow the direction of steepest descent of the associated free energy functional [106] at each instant of time, rendering a process where the entropy is maximized, i.e., a diffusion.

In Section 2.4.1 we set dynamics for the stochastic particle flow. Section 2.4.2 derives the stationary solution to the Fökker-Planck equation such that the particles follow the Langevin dynamics. In Section 2.4.3 we show how to specify the Langevin dynamics to solve the specific problem of interest. In Section 2.4.4 we discuss the interpretation of and possible choices for the diffusion matrix; in Section 2.4.5 we present the integration methods used to sample from the Langevin dynamics; in Section 2.4.6 we discuss criteria for choosing the algorithm's parameters (the step size and time horizon).

2.4.1 Dynamics of Particles

Assuming that a set of particles $\{\mathrm{x}^{(i)}(\lambda)\}_{i\in[1..N]}$ follows a diffusion process $\{\mathrm{X}_\lambda\}_{\lambda\geq 0}$ when subject to a Bayesian measurement update, the dynamics of the particles can, in general, be described by the Îto stochastic differential equation

$$d\mathrm{X}_\lambda = \mu(\mathrm{X}_\lambda,\lambda)d\lambda + \sigma(\mathrm{X}_\lambda,\lambda)d\mathrm{W}_\lambda, \quad \mathrm{X}_0 = \mathrm{X}(0)\,; \tag{2.11}$$

such that the associated probability distribution, $p(\mathrm{x},\lambda)$, is continuously evolving with respect to the pseudo-time variable $\lambda\in\mathbb{R}_+$, where $\{\mathrm{W}_\lambda\}_{\lambda\geq 0}$ is a standard Brownian motion, $\mu(\mathrm{X}_\lambda,\lambda)$ is the drift vector and $\sigma(\mathrm{X}_\lambda,\lambda)$ is the diffusion coefficient. It is well known [101, 78] that the probability density $p(\mathrm{x},\lambda)$ of an d_x-dimensional random state vector x under the dynamics of (2.11) has a deterministic evolution according to the Fökker-Planck equation

$$\begin{aligned}\frac{\partial}{\partial\lambda}p(\mathrm{x},\lambda) = &-\sum_{i=1}^{d_x}\frac{\partial}{\partial x_i}\left[\mu_i(\mathrm{x},\lambda)\,p(\mathrm{x},\lambda)\right]\\ &+\frac{1}{2}\sum_{i=1}^{d_x}\sum_{j=1}^{d_x}\frac{\partial}{\partial x_i}\frac{\partial}{\partial x_j}\left[\mathrm{D}_{ij}(\mathrm{x},\lambda)\,p(\mathrm{x},\lambda)\right],\\ &p(\mathrm{x},0) = p_0(\mathrm{x}),\ \lambda\geq 0;\end{aligned} \tag{2.12}$$

where $\mathrm{x} = [x_1,\ldots,x_{d_x}]^T$, $\mu = [\mu_1,\ldots,\mu_{d_x}]^T$, and

$$\mathrm{D}_{ij}(\mathrm{x},\lambda) = \sum_{k=1}^{d_x}\sigma_{ik}(\mathrm{x},\lambda)\,\sigma_{jk}(\mathrm{x},\lambda), \tag{2.13}$$

for an d_x-dimensional Wiener process $\{\mathrm{W}_\lambda\}_{\lambda\geq 0}$. In its usual form, as described in Physics, the equation reads

$$\frac{\partial}{\partial\lambda}p = -\nabla_{\mathrm{x}}\cdot[\mu p] + \frac{1}{2}\nabla_{\mathrm{x}}\cdot[\mathrm{D}\nabla_{\mathrm{x}}p]\,. \tag{2.14}$$

We assume that the diffusion coefficient σ is locally independent of x, giving rise to a local diffusion matrix $\mathrm{D}(\lambda) = \sigma(\lambda)\,\sigma(\lambda)^T$ that is invariant to the divergence operator in the vicinity of each particle. This means that, at a given time instant, the diffusion term in (2.14) evolves at a rate proportional to the curvature of a (Riemann) manifold that is approximately constant in the neighborhood of each particle. This assumption does not affect the generality of the concepts applied in our derivation for two reasons: it results in a stochastic particle flow that is missing a simple term, of the form $\sigma(x,\lambda)\cdot\partial_x[\sigma(x,\lambda)]$, that could be incorporated if needed; in practice, any probability density can be well approximated by a mixture of densities whose covariances are locally constant with respect to the state [177] (i.e., $\partial_x[\sigma(x,\lambda)] = 0$ locally). Additionally, as evidenced in [80], keeping the diffusion coefficient fixed for each sampling step does not perturb the target distribution.

2.4.2 Stationary Solution of the Fökker-Planck Equation

A stationary solution to the equation (2.14) should satisfy

$$\frac{\partial}{\partial\lambda}p(\mathrm{x},\lambda) \overset{\lambda\to\infty}{\longrightarrow} 0. \tag{2.15}$$

By writing

$$\nabla_{\mathrm{x}}\cdot S \triangleq \nabla_{\mathrm{x}}\cdot[\mu\,p] - \frac{1}{2}\nabla_{\mathrm{x}}\cdot[\mathrm{D}\nabla_{\mathrm{x}}p], \tag{2.16}$$

the probability current S is identified as

$$\begin{aligned} S(\mathrm{x},\lambda) &\triangleq \mu(\mathrm{x},\lambda)\,p(\mathrm{x},\lambda) - \frac{1}{2}\mathrm{D}(\lambda)\cdot\nabla_{\mathrm{x}}p(\mathrm{x},\lambda) \\ &= p(\mathrm{x},\lambda)\left[\mu(\mathrm{x},\lambda) - \frac{1}{2}\mathrm{D}(\lambda)\cdot\nabla_{\mathrm{x}}\log p(\mathrm{x},\lambda)\right]. \end{aligned} \tag{2.17}$$

Since the stationary condition requires

$$\frac{\partial}{\partial\lambda}p(\mathrm{x},\lambda) = -\nabla_{\mathrm{x}}\cdot S(\mathrm{x},\lambda) \overset{\lambda\to\infty}{\longrightarrow} 0, \tag{2.18}$$

the probability current is required to vanish as $\lambda\to\infty$. The probability current can only vanish if the drift $\mu(\mathrm{x},\lambda)$ can be expressed as the gradient of a potential function [165], which induces the terms within brackets in (2.17) to cancel out. We write the drift as the gradient of a stationary potential function according to

$$\mu(\mathrm{x},\lambda) = -\frac{1}{2}\mathrm{D}(\lambda)\cdot\nabla_{\mathrm{x}}\Phi(\mathrm{x}). \tag{2.19}$$

The potential conditions

$$\frac{\partial\mu_i}{\partial x_j} = \frac{\partial\mu_j}{\partial x_i}, \quad \forall i\neq j, \tag{2.20}$$

are necessary and sufficient for the existence of $\Phi(\mathrm{x})$ [165]. Provided that the probability current vanishes as $\nabla_{\mathrm{x}}\log p(\mathrm{x},\lambda)\to-\nabla_{\mathrm{x}}\Phi(\mathrm{x})$, we obtain the stationary solution, $p_{\mathrm{st}}(\mathrm{x})$, as

$$p(\mathrm{x},\lambda) \overset{\lambda\to\infty}{\longrightarrow} p_{\mathrm{st}}(\mathrm{x}) = \frac{1}{Z}e^{-\Phi(\mathrm{x})}, \tag{2.21}$$

where

$$Z = \int_{\mathbb{R}^{d_x}} e^{-\Phi(\mathrm{x})} d\mathrm{x} \tag{2.22}$$

must be positive and finite. We promptly recognize (2.21) as analogous to the Gibbs distribution. It is verifiable that (see, for example, [105]) the Gibbs distribution minimizes the free energy functional over all probability densities on $\mathbb{R}^{d_x}$. It can also be shown that the stationary solution is the first eigenfunction of the Fökker-Planck equation, corresponding to the eigenvalue zero [165].

2.4.3 The Stochastic Flow

The general stochastic particle flow is derived by setting the stationary solution, $p_{st}(\mathrm{x})$, to be the target posterior density, $\pi(\mathrm{x}) = p(\mathrm{x}|\mathrm{y}_{1:k})$, to give

$$\begin{aligned} p_{\mathrm{st}}(\mathrm{x}) &\coloneqq p(\mathrm{x}|\mathrm{y}_{1:k}), \\ \frac{e^{-\Phi(\mathrm{x})}}{Z} &= \frac{p(\mathrm{y}_k|\mathrm{x})\, p(\mathrm{x}|\mathrm{y}_{1:k-1})}{p(\mathrm{y}_k|\mathrm{y}_{1:k-1})}, \\ \Phi(\mathrm{x}) &= -\log p(\mathrm{y}_k|\mathrm{x}) - \log p(\mathrm{x}|\mathrm{y}_{1:k-1}). \end{aligned} \tag{2.23}$$

Given a valid potential function $\Phi(\mathrm{x})$ provides the stationary solution, all potential functions of the form $\Phi(\mathrm{x}) \pm K$ for any constant $K \in \mathbb{R}$ are also valid. We can therefore choose a valid potential function in (2.23) such that $p(\mathrm{y}_k|\mathrm{y}_{1:k-1}) = Z$. By using equation (2.19), we obtain

$$\begin{aligned} \mu(\mathrm{x}, \lambda) &= -\frac{1}{2}\mathrm{D}(\lambda) \cdot \nabla_{\mathrm{x}} \Phi(\mathrm{x}) = \frac{1}{2}\mathrm{D}(\lambda) \cdot \nabla_{\mathrm{x}} \log \pi(\mathrm{x}) \\ &= \frac{1}{2}\mathrm{D}(\lambda) \cdot \left[\nabla_{\mathrm{x}} \log p(\mathrm{y}_k|\mathrm{x}) + \nabla_{\mathrm{x}} \log p(\mathrm{x}|\mathrm{y}_{1:k-1})\right]. \end{aligned} \tag{2.24}$$

Substituting (2.24) into (2.17), it is easy to see that the probability current vanishes as $\lambda \to \infty$. Additionally, it is important to note that continuous multivariate probability densities commonly used in parametric statistics (e.g., Gaussian, Student's t, Mises-Fisher, Pareto of first kind, Cauchy etc) satisfy the potential conditions (2.20) that suffice for $\Phi(\mathrm{x})$ to exist.

Based on equation (2.11) and on the drift obtained from the stationary solution (2.24), the dynamics of a set of particles $\{\mathrm{x}^{(i)}(\lambda)\}_{i\in[1..N]}$ can be described by the stochastic differential equation

$$\begin{aligned} d\mathrm{x} &= \mu(\mathrm{x}, \lambda)\, d\lambda + \sigma(\mathrm{x}, \lambda)\, d\mathrm{w}_\lambda, \quad \mathrm{x}_0^{(i)} = \mathrm{x}_{k-1}^{(i)}; \\ d\mathrm{x} &= \frac{1}{2}\mathrm{D}\nabla_{\mathrm{x}} \log \pi(\mathrm{x})\, d\lambda + \mathrm{D}^{1/2} d\mathrm{w}_\lambda; \end{aligned} \tag{2.25}$$

where $\pi(\mathrm{x})$ is the target (posterior) probability density, $\{\mathrm{w}_\lambda\}_{\lambda\geq 0}$ is the standard Wiener process (Brownian motion) and D is the diffusion matrix. The stochastic process described by (2.25) is known in the literature to follow the Langevin dynamics and, except for few special cases, cannot be exactly simulated. Thus, the most common way to ensure the simulation provides samples from the correct target distribution, $\pi(\mathrm{x})$, is to set the discretized dynamics as a proposal within a Markov chain Monte Carlo framework, which leads to the Metropolis-adjusted Langevin algorithm (MALA) [168].

By defining the distribution at λ and target distribution as $d\mathcal{P} = p(\mathrm{x}, \lambda)\, d\mathrm{x}$ and $d\mathcal{P}_\pi = \pi(\mathrm{x})\, d\mathrm{x}$

respectively, one can articulate the total-variation distance between the probability measures $\mathcal{P}(dx)$ and $\mathcal{P}_\pi(dx)$ defined on $\left(\mathbb{R}^{d_x}, \mathscr{B}(\mathbb{R}^{d_x})\right)$[6] as $\|\mathcal{P} - \mathcal{P}_\pi\|_{\text{TV}}$. It can be shown that (e.g., [41]), if the SDE (2.25) is integrated over a sufficiently long (finite) time horizon $T \in \mathbb{R}_+$, then[7]

$$\|\mathcal{P} - \mathcal{P}_\pi\|_{\text{TV}} = \frac{1}{2} \int_{\mathbb{R}^{d_x}} |p(\mathrm{x}, \lambda) - \pi(\mathrm{x})| \, d\mathrm{x} \leq \varepsilon \tag{2.26}$$

for any $\lambda > T$, under a desired precision ε. The implication is that the stochastic particle flow implements the filtering mapping (2.4) with increasing accuracy as λ progresses.

The stochastic particle flow can be interpreted as a continuous-time filtering method in the classical sense. Under the abstraction of a continuously interpolated observation process, the method has a direct correspondence to the Kallianpur-Striebel formula and satisfies the Zakai equation as we demonstrate by the *Theorem 2.17* and *Corollary 2.19* in Section 2.9.2. Practically speaking, the major difference between the stochastic particle flow and other Langevin-based algorithms is the way that the target density is sequentially approximated via local representations that compose a mixture. This will be discussed later in Section 2.5.

2.4.4 The Diffusion Matrix

As explained by Girolami & Calderhead [80], the space of parameterized probability density functions is endowed with a natural Riemann geometry, where the diffusion matrix arises as the inverse of a position-specific metric tensor, $\mathrm{G}(\mathrm{x}(\lambda))$. This metric tensor maps the distances inscribed in a Riemann manifold to distances in the Euclidean space and, therefore, constitutes a means to constrain the dynamics of any stochastic process to the geometric structure of the parametric probability space. Rao [159] showed the tensor $\mathrm{G}(\mathrm{x}(\lambda))$ to be the expected Fisher information matrix

$$\mathrm{G}(\mathrm{x}(\lambda)) = -\mathbb{E}_{\mathrm{y}|\mathrm{x}}\left[\mathcal{H}_{\mathrm{x}}\left[\log p(\mathrm{y}|\mathrm{x})\right]\right], \tag{2.27}$$

where $\mathcal{H}_{\mathrm{x}}[.]$ is the Hessian matrix with respect to x. In a Bayesian context, Girolami & Calderhead [80] suggested a metric tensor that includes the prior information as

$$\mathrm{G}(\mathrm{x}(\lambda)) = -\mathbb{E}_{\mathrm{y}|\mathrm{x}}\left[\mathcal{H}_{\mathrm{x}}\left[\log p(\mathrm{y}|\mathrm{x})\right]\right] - \mathcal{H}_{\mathrm{x}}\left[\log p_x(\mathrm{x})\right], \tag{2.28}$$

although many possible choices of metric for a specific manifold could be advocated. Because we are interested in local (curvature) properties of the stochastic particle flow, a sensible choice for the metric tensor $\mathrm{G}(\mathrm{x}(\lambda))$ is the observed Fisher information matrix incorporating the prior information. In this case, the diffusion matrix becomes

$$\mathrm{D} = \mathrm{G}(\mathrm{x}(\lambda))^{-1} = \left[-\mathcal{H}_{\mathrm{x}}\left[\log \pi(\mathrm{x})\right]\right]^{-1}_{\mathrm{x}=\mathrm{x}_\lambda}, \tag{2.29}$$

where the Hessian matrix is locally evaluated at $\mathrm{x} = \mathrm{x}_\lambda^{(i)}$ for the ith sample, and the resulting diffusion matrix is kept constant for each integration step to obtain the subsequent sample. A problem with this choice is that the expression (2.29) may not be strictly positive definite at specific points of the state space for some types of probability distributions (e.g., mixtures). In order to solve that problem, one could appeal to methods for regularizing the diffusion matrix

[6] $\mathscr{B}(\mathbb{R}^{d_x})$ is the σ-field composed of the Borel sets of $\mathbb{R}^{d_x}$.

[7] The total variation norm for probability measures have an equivalence to the $\mathbb{L}^1$-norm as presented in (2.26). A simple argument for this equivalence is given in [119], chapter 4, proposition 4.2.

such as the Tikhonov regularization, the technique to find the nearest (in terms of minimum Fröbenius norm) positive definite matrix [92], or *SoftAbs* [13], a technique that implements a smooth absolute transformation of the eigenvalues to map the negative-Hessian metric into a positive-definite matrix. Another possibility is adopting an empirical estimate to (2.28).

2.4.5 Integration Method

Among the discretization methods that could be used to integrate the SDE (2.25), we advocate the use of Ozaki's discretization [152] of the Langevin diffusion. This is more accurate than methods based on the Euler discretization. Ozaki's discretization is only possible for target densities that are continuously differentiable and have a smooth Hessian matrix. These requirements may be fulfilled by a solution that constitutes a superposition of conveniently parameterized local approximations to a density.

The algorithm that enables simulation from the SDE (2.25) using Ozaki's discretization is generally called *Langevin Monte Carlo with Ozaki discretization* (LMCO) in the MCMC community (see [41]). Provided an appropriate time horizon, T, by discretizing the interval $0 \leq \lambda \leq T$ into L sub-intervals $\{\lambda_0 = 0, \lambda_1, \ldots \lambda_l, \ldots, \lambda_L = T\}$, the discretized particle flow equation using Ozaki's method is given by

$$\begin{aligned} \mathrm{x}(\lambda_{l+1}) = \mathrm{x}(\lambda_l) &+ \left(\mathbb{I}_{d_x} - e^{-\frac{\Delta\lambda}{2}\mathrm{D}(\lambda_l)^{-1}}\right) \mathrm{D}(\lambda_l)^2 \nabla_{\mathrm{x}} \log \pi\left(\mathrm{x}(\lambda_l)\right) \\ &+ \left[\left(\mathbb{I}_{d_x} - e^{-\Delta\lambda \mathrm{D}(\lambda_l)^{-1}}\right) \mathrm{D}(\lambda_l)^2\right]^{1/2} \mathrm{w}_{l+1}, \end{aligned} \tag{2.30}$$

where $\{\mathrm{w}_l : l = 1, \ldots, L\}$ is a sequence of independent random vectors distributed according to $\mathrm{w}_l \sim \mathcal{N}(\mathrm{w}; 0_{d_x}, \mathbb{I}_{d_x})$. The need to compute the exponential matrices in (2.30) implies an increment in complexity typically bounded by $\mathcal{O}(NLd_x^3)$ computations, which may not be justifiable for some applications. A cheaper alternative is achieved by linearizing (2.25) in the neighborhood of the current state, assuming $\mathrm{D}(\lambda)$ piecewise constant in pseudo-time, transforming the linearized equation by the Laplace transform, solving it in the Laplace domain, and transforming it back. The result is

$$\begin{aligned} \mathrm{x}(\lambda_{l+1}) = \mathrm{x}(\lambda_l) &+ \left(1 - e^{-\frac{1}{2}\Delta\lambda}\right) \mathrm{D}(\lambda_l) \nabla_{\mathrm{x}} \log \pi\left(\mathrm{x}(\lambda_l)\right) \\ &+ \left(1 - e^{-\Delta\lambda}\right)^{1/2} \mathrm{D}(\lambda_l)^{1/2} \mathrm{w}_{l+1}, \end{aligned} \tag{2.31}$$

where the exponential matrices are avoided but the exponential effect on the integration variable (time step) is kept. See Section 2.9.3 for the derivation of this latter integration rule.

It is important to note that, upon integrating the SDE (2.25) by a numerical method, convergence to the invariant distribution is no longer guaranteed for any finite step size. This is due to the first-order integration error that is introduced. When tackling difficult nonlinear filtering problems where the integration error becomes significant, a correction can be carried out by employing a Metropolis acceptance step after each integration step to ensure convergence to the invariant measure.

2.4.6 Selection of Time Horizon and Integration Step Size

To successfully implement the stochastic particle flow, it is necessary to select an adequate time horizon, T, and an integration step size, $\Delta\lambda$. These parameters need to be chosen such that stationarity is reached and convergence to the invariant measure is achieved. There are several routes one could take to solve this problem with each making different assumptions about the probability measures involved and about the regularity properties of the stationary distribution. One could also pose related questions in the context of specific implementations. Answering such questions might, for example, involve selecting the time horizon and integration step size that minimizes the variance of the samples' weights.

The view adopted here is that, since computational effort is a fundamental issue for implementing the stochastic particle flow, we should choose these parameters to minimize computational effort. More specifically, we want to minimize the number of integration steps that need to be performed to achieve

$$\|\mathcal{P}_{\tilde{\mathcal{L}}[\Delta\lambda],T} - \mathcal{P}_{\pi}\|_{\mathrm{TV}} \leq \varepsilon \tag{2.32}$$

for an acceptable precision level ε, where $\mathcal{P}_{\tilde{\mathcal{L}}[\Delta\lambda],T}(d\mathrm{x})$ is the approximating probability measure achieved by sampling from the discretized Langevin stochastic process over $L = \lceil T/\Delta\lambda \rceil$ steps.

Defining near-optimal choices of these parameters for general target measures, including mixtures and highly skewed distributions, would require a more thorough study that extrapolates the scope of this work. Instead, in this chapter, we propose two pragmatic approaches to choosing both the time horizon and integration step size.

Approach 1

Our first approach builds on results concerning the scaling of Langevin-based MCMC algorithms: the interested reader is referred to Roberts & Rosenthal [169] and a recent extension by Pillai *et al.* [155] that treat high-dimensional target measures that are not of the product form. In summary, these analyses show that the number of steps required to sample the target measure by the Metropolis-Adjusted Langevin algorithm (MALA) grows as $L \sim \mathcal{O}(d_x^{1/3})$. In addition, these papers work out the optimal step size by maximizing the "speed function" or, equivalently, producing the optimal average acceptance rate. Although this optimal criterion is only applicable to algorithms that employ a Metropolis acceptance step, tuning the step size used in the stochastic particle flow for an "emulated" (hypothetical) acceptance rate of interest can guide the rate of convergence (even if the accept-reject step is suppressed in practice). Abusing the methodology presented by Pillai *et al.* [155] and using *Proposition 2.4* from Roberts *et al.* [167], let us denote the asymptotic acceptance probability $\alpha(\ell)$ as a function of a scaling parameter $\ell \in \mathbb{R}$, such that the speed function $\mathfrak{h}(\ell)$ for high-dimensional MALA can be approximated as [155]

$$\begin{aligned}
\mathfrak{h}(\ell) &= \ell \cdot \alpha(\ell) \approx \ell \cdot \mathbb{E}_{\pi}\left[1 \wedge e^{\mathcal{N}\left(-\ell^3/4, \ell^3/2\right)}\right] \\
&= \ell \cdot \left[\mathcal{N}_{\mathrm{cdf}}\left(\frac{-\ell^3/4}{\sqrt{\ell^3/2}}\right) + e^{-\frac{\ell^3}{4} + \frac{\left((\ell^3/2)^{1/2}\right)^2}{2}} \mathcal{N}_{\mathrm{cdf}}\left(-\sqrt{\ell^3/2} - \frac{-\ell^3/4}{\sqrt{\ell^3/2}}\right)\right] \\
&= \ell \cdot 2\mathcal{N}_{\mathrm{cdf}}\left(-\sqrt{\ell^3/8}\right),
\end{aligned} \tag{2.33}$$

where $\mathcal{N}_{\text{cdf}}(.)$ is a standard normal cumulative distribution function (cdf). One can observe the maximum occurs at $\ell_{\text{opt}} \approx 1.3620$ and corresponds to the theoretical optimal acceptance rate, $\alpha(\ell_{\text{opt}}) \approx 0.5741$.

Our approach is then based on the notion that, for a conveniently selected acceptance rate, the step size should scale as $\Delta\lambda \propto \ell \cdot d_x^{-\xi}$ such that the total number of steps is scaled as $L \propto d_x^{\xi}$, and the exponent ξ depends on whether the Metropolis adjustment is used or not. Theoretically, if the accept-reject step is not adopted then[8] $\xi = 1$, otherwise the optimal choice is $\xi = 1/3$ [155]. To exemplify the use of this approach, to produce an emulated (asymptotic) acceptance rate of $\alpha = 0.80$, a stochastic particle flow should be scaled as

$$\ell \approx 2\left[-\mathcal{N}_{\text{cdf}}^{-1}(\alpha/2)\right]^{2/3} = 0.8008.$$

Using the Langevin dynamics considered herein, if $d_x = 10$ and $\xi = 1$ then $\Delta\lambda = 2\ell \cdot d_x^{-\xi} \approx 0.1602$.

If the accept-reject step is present, a criterion to stop the simulation could be established online. Various MCMC convergence diagnostic methods are applicable to this task. However, in our experience with the stochastic particle flow, such approaches to online determination of convergence may indicate more steps are needed than are actually necessary to obtain good results, and so give a pessimistic view of the amount of computation required. To set the time horizon when accept-reject step is not used, we note that $T = T_0 + L_s \cdot \Delta\lambda$, where T_0 is the time required to take the chain to the region of high acceptance probability ("warm-up") and L_s is the number of steps to explore the invariant measure. We determine both L_s and T_0 either by presetting reasonable values, or based on the second approach to be explained next.

Approach 2

Our second approach is an extension of the method proposed by Dalalyan [41]. It is useful due to its ease of application and suitability to "nicely" measurable filtering quantities although, strictly speaking, the method is only applicable to target densities that are log-concave. The criteria are presented as follows.

Theorem 2.1. *Let $\Phi : \mathbb{R}^{d_x} \to \mathbb{R}$ be a measurable convex function satisfying*

$$\int_{\mathbb{R}^{d_x}} \exp\{-\Phi(\mathrm{x})\} < \infty, \tag{2.34}$$

$$\Phi(\mathrm{x}) - \Phi(\bar{\mathrm{x}}) - \nabla_{\mathrm{x}}\Phi(\bar{\mathrm{x}})^T(\mathrm{x} - \bar{\mathrm{x}}) \geq \frac{1}{2} m \|\mathrm{x} - \bar{\mathrm{x}}\|_2^2, \tag{2.35}$$

$$\|\nabla_{\mathrm{x}}\Phi(\mathrm{x}) - \nabla_{\mathrm{x}}\Phi(\bar{\mathrm{x}})\|_2 \leq M \|\mathrm{x} - \bar{\mathrm{x}}\|_2, \quad \forall \mathrm{x}, \bar{\mathrm{x}} \in \mathbb{R}^{d_x}, \tag{2.36}$$

for two existing positive constants m and M. Let $\bar{\mathrm{x}} \in \mathbb{R}^{d_x}$ be the global minimum of $\Phi(\mathrm{x})$. Suppose a discrete-time Langevin Monte Carlo algorithm integrates (2.25), targeting the invariant density $\pi(\mathrm{x}) \propto \exp\{-\Phi(\mathrm{x})\}$ with measure $\mathcal{P}_\pi(d\mathrm{x})$, and with the initial density $\nu(\mathrm{x}) = \delta(\mathrm{x} - \mathrm{x}_\nu)$ (a probability mass initially located at $\mathrm{x} = \mathrm{x}_\nu$). In addition, assume that for some $\gamma \geq 1$ we have $\Delta\lambda \leq (\gamma M)^{-1}$, and $K = \sup_{\mathrm{x}} \|\mathrm{D}(\mathrm{x})\|_2$ where $\mathrm{D}_\lambda = \mathrm{D}(\mathrm{x}_\lambda)$ is the diffusion matrix. Then, for a time horizon, T, and step size, $\Delta\lambda$, the total-variation distance between the target measure $\mathcal{P}_\pi$ and the approximated measure $\mathcal{P}_{\tilde{\mathcal{L}}(\Delta\lambda),T}$ furnished by the discrete-time Langevin Monte Carlo

[8] In view of (2.38) and (2.39), $T/\Delta\lambda \sim \mathcal{O}(d_x)$.

algorithm satisfies

$$\begin{aligned}\|\mathcal{P}_{\tilde{\mathcal{L}}[\Delta\lambda],T} - \mathcal{P}_{\pi}\|_{TV} \\ \leq \frac{1}{2}\exp\left\{-\frac{1}{2}mT + \frac{d_x}{2}\log\left(\frac{M}{m}\right) - \log\left[\Gamma_u\left(\frac{d_x}{2}, \frac{M\|\bar{\mathrm{x}} - \mathrm{x}_\nu\|_2^2}{2}\right)\right]\right\} + \frac{1}{2} \\ - \frac{1}{2}\exp\left\{-\frac{d_x}{2}\frac{M^3K^4\gamma}{48(2\gamma-1)}\left(\frac{1}{d_x}\|\bar{\mathrm{x}} - \mathrm{x}_\nu\|_2^2 + 2T\right)\Delta\lambda^2 - \frac{d_x M^2 K^3 T}{16}\Delta\lambda\right\}, \quad (2.37)\end{aligned}$$

where $\Gamma_u(s,x) \triangleq \Gamma(s)^{-1}\int_x^\infty t^{s-1}e^{-t}dt$ *is the upper incomplete gamma function.*

Corollary 2.2. *Let* $d_x \geq 2$, Φ *satisfy (2.34), (2.35) and (2.36), and* $\varepsilon \in (0, 1/2)$ *be a desired precision level. Let the time horizon* T *and the step size* $\Delta\lambda$ *be defined by*

$$T \geq \frac{2\log(1/\varepsilon) + d_x\log\left(\frac{M}{m}\right) - 2\log\left[\Gamma_u\left(\frac{d_x}{2}, \frac{M\|\bar{\mathrm{x}}-\mathrm{x}_\nu\|_2^2}{2}\right)\right]}{m}, \quad (2.38)$$

$$\Delta\lambda \leq \frac{-\frac{T}{16} + \sqrt{\left(\frac{T}{16}\right)^2 + \frac{\gamma}{48(2\gamma-1)}\left(\frac{1}{d_x}\|\bar{\mathrm{x}} - \mathrm{x}_\nu\|_2^2 + 2T\right)M^{-1}K^{-2}\left[\frac{2}{d_x}\log\left(\frac{1}{1-\varepsilon}\right)\right]}}{\frac{\gamma}{48(2\gamma-1)}\left(\frac{1}{d_x}\|\bar{\mathrm{x}} - \mathrm{x}_\nu\|_2^2 + 2T\right)MK}, \quad (2.39)$$

where $\gamma \geq 1$. *Then the resulting probability distribution of a Langevin Monte Carlo algorithm that integrates (2.25) after* $L = \lceil T/\Delta\lambda \rceil$ *steps, satisfies* $\|\mathcal{P}_{\tilde{\mathcal{L}}[\Delta\lambda],T} - \mathcal{P}_{\pi}\|_{TV} \leq \varepsilon$.

Theorem 2.1 is thoroughly underpinned by the findings of Dalalyan [43], with specific settings changed to match the Langevin algorithm proposed in this chapter, and both a more general bound for the time horizon and a tightened bound for the step size (to reduce computational effort). We recommend the reader interested in the proof to first refer to [43] and then follow the missing arguments for its proof in Section 2.9.1.

Corollary 2.2 is a direct criterion for selecting the time horizon and step size, arising from the right-hand side of the inequality (2.37) being set to be a desired precision level. It is essential to clarify that some practical issues arise here: in this form, the method holds for log-concave densities only; the positive constants, m and M, are assumed known a priori; and the approximation of the filtering density is not taken into account in the error budget. Rigorously speaking, the method does not apply to more general cases. However, the method has utility as the basis of an approximate (and pragmatic) mechanism for obtaining the required parameters. In making this approximation, we explicitly acknowledge that we are assuming that:

1. The target density can be well characterized by a central tendency statistic, $\bar{\mathrm{x}}_c$, that replaces and roughly represents $\bar{\mathrm{x}}$ in all aspects of the analysis.

2. The initial measure is composed of a superposition of N probability masses described by

$$N^{-1}\sum_{i=1}^{N}\delta(\mathrm{x} - \mathrm{x}^{(i)})$$

or, equivalently, an empirical distribution with mean μ_ν and covariance matrix V, spatially encompassing all initial samples (from the previous filtering iteration), which is assumed

to constrain the constants M and m by

$$M < \|\mathrm{V}\|_2/2, \tag{2.40}$$

$$(\mathrm{x}^{(i)} - \bar{\mathrm{x}}_c)^T \frac{m}{2}(\mathrm{x}^{(i)} - \bar{\mathrm{x}}_c) \geq \chi^{-2}_{\mathrm{cdf}}(0.99, d_x), \quad \forall \mathrm{x}^{(i)} : i = 1, \ldots, N, \tag{2.41}$$

where $\chi^{-2}_{\mathrm{cdf}}(P, \kappa)$ is the inverse of chi-square cdf for probability P and κ degrees of freedom.

3. In accordance with *Lemma 4* in [41], the constant M is also constrained by

$$\Phi(\mathrm{x}^{(i)}) - \Phi(\bar{\mathrm{x}}_c) - \nabla_{\mathrm{x}}\Phi(\bar{\mathrm{x}}_c)^T(\mathrm{x}^{(i)} - \bar{\mathrm{x}}_c) \leq \frac{M}{2}\left\|\mathrm{x} - \bar{\mathrm{x}}_c\right\|_2^2, \quad \forall \mathrm{x}^{(i)} : i = 1, \ldots, N. \tag{2.42}$$

4. The positive constants M and m can be roughly estimated by

   (a) approximating the statistic $\bar{\mathrm{x}}_c$ of the target density (e.g., obtaining a maximum-a-posteriori estimate by optimization or an approximated mean by the EKF),

   (b) inverting conditions (2.35) and (2.36), and incorporating the constraint (2.42), to give

$$\tilde{M} = 2\sup_{i\in[1..N]} \max\left[\frac{\left\|\nabla_{\mathrm{x}}\Phi(\mathrm{x}^{(i)}) - \nabla_{\mathrm{x}}\Phi(\bar{\mathrm{x}}_c)\right\|_2}{2\left\|\mathrm{x}^{(i)} - \bar{\mathrm{x}}_c\right\|_2}, \frac{\Phi(\mathrm{x}^{(i)}) - \Phi(\bar{\mathrm{x}}_c) - \nabla_{\mathrm{x}}\Phi(\bar{\mathrm{x}}_c)^T(\mathrm{x}^{(i)} - \bar{\mathrm{x}}_c)}{\left\|\mathrm{x}^{(i)} - \bar{\mathrm{x}}_c\right\|_2^2}\right], \tag{2.43}$$

$$\tilde{m} = 2\inf_{i\in[1..N]} \frac{\Phi(\mathrm{x}^{(i)}) - \Phi(\bar{\mathrm{x}}_c) - \nabla_{\mathrm{x}}\Phi(\bar{\mathrm{x}}_c)^T(\mathrm{x}^{(i)} - \bar{\mathrm{x}}_c)}{\left\|\mathrm{x}^{(i)} - \bar{\mathrm{x}}_c\right\|_2^2}, \tag{2.44}$$

   where all quantities can be computed from $\Phi(\mathrm{x}) \triangleq -\log p(\mathrm{y}_k|\mathrm{x}) - \log p(\mathrm{x}|\mathrm{y}_{1:k-1})$ given an approximation to the prior pdf; the resulting values of $\tilde{M}$ and $\tilde{m}$ must also satisfy (2.40) and (2.41).

Once the positive constants M and m have been estimated, obtaining T and $\Delta\lambda$ follows from (2.38) and (2.39) respectively. In our experience, for very simple problems, (2.38) may produce overestimated time horizons and, as a consequence, may cause (2.39) to produce underestimated step sizes for the stochastic particle flow. This happens because the bound for the time horizon becomes loose, in view of *Lemma 2.7*, for initial distributions that are far from the target distribution. In simple cases, a closer approximation can by achieved by assuming 1-uniform ergodicity of the Markov chain to give

$$T \geq \frac{2\log(1/\varepsilon) + d_x \log R}{\tilde{m}}, \tag{2.45}$$

for a finite $R \in \mathbb{R}_+$, at the cost of having to determine R empirically. Similarly, for simple cases, expression (2.39) can be replaced with an empirical rule of the form

$$\Delta\lambda \leq 2\frac{\sqrt{\tilde{m}}}{\tilde{M}}, \tag{2.46}$$

which satisfies $\Delta\lambda \leq \left(\gamma\tilde{M}\right)^{-1}$ for $\gamma \leq \left(2\sqrt{\tilde{m}}\right)^{-1}$ as required by *Theorem 2.1* but is not guaranteed to satisfy *Corollary 2.2.*

2.5 Stochastic Particle Flow as a Gaussian Sum Filter

In this section, we use the stochastic particle flow to derive a filter that approximates the posterior probability density as a Gaussian mixture. We refer to the resulting filter as the stochastic particle flow Gaussian sum filter (SPF-GS).

2.5.1 The Mixture-Based Approximating Measure

Given a set of samples drawn from an importance distribution, $\{\mathrm{x}^{(i)} \in \mathcal{X}\}_{i\in[1..N]}$, if one is required to solve the filtering problem by a standard Monte Carlo method, then the stochastic filter adopts the following approximation

$$\begin{aligned}\hat{\varphi} &= \int_{\mathcal{X}} \varphi(\mathrm{x})\,\pi(\mathrm{x})\,dx \\ &\approx \int_{\mathcal{X}} \varphi(\mathrm{x}) \sum_{i=1}^{N} w(\mathrm{x}^{(i)})\delta(\mathrm{x}-\mathrm{x}^{(i)})d\mathrm{x} \\ &= \sum_{i=1}^{N} w^{(i)}\varphi(\mathrm{x}^{(i)}),\end{aligned} \tag{2.47}$$

where $w(\mathrm{x}^{(i)}) = w^{(i)}$ are the importance weights. Now suppose that we have access to an approximating measure $\tilde{\mathcal{P}}_\pi(d\mathrm{x})$ on $(\mathcal{X}, \mathscr{B}(\mathcal{X}))$ with an associated density such that $d\tilde{\mathcal{P}}_\pi = \tilde{\pi}d\mathrm{x}$. If the density $\tilde{\pi}$ involves a mixture of N Gaussians according to

$$\begin{aligned}\tilde{\pi}(\mathrm{x}) &= \sum_{i=1}^{N} w_m^{(i)}\mathcal{N}(\mathrm{x};\mu_m(\mathrm{x}^{(i)}),\Sigma_m(\mathrm{x}^{(i)})) \\ &= \sum_{i=1}^{N} w_m^{(i)}\mathcal{N}(\mathrm{x};\mu_m^{(i)},\Sigma_m^{(i)}),\end{aligned} \tag{2.48}$$

where $\{w_m^{(i)}, \mu_m^{(i)}, \Sigma_m^{(i)}\}_{i\in[1..N]}$ are computed based on the samples $\{\mathrm{x}^{(i)}\}_{i\in[1..N]}$, then the solution is given by

$$\begin{aligned}\hat{\varphi} &= \int_{\mathcal{X}} \varphi(\mathrm{x})\,\pi(\mathrm{x})\,d\mathrm{x} \\ &\approx \int_{\mathcal{X}} \varphi(\mathrm{x}) \sum_{i=1}^{N} w_m^{(i)}\mathcal{N}(\mathrm{x};\mu_m^{(i)},\Sigma_m^{(i)})d\mathrm{x} \\ &= \sum_{i=1}^{N} w_m^{(i)} \int_{\mathcal{X}} \varphi(\mathrm{x})\,\mathcal{N}(\mathrm{x};\mu_m^{(i)},\Sigma_m^{(i)})d\mathrm{x} \\ &= \sum_{i=1}^{N} w_m^{(i)}\mathbb{E}_{\mathcal{N}}\left[\varphi(\mathrm{x})\,|\mu_m^{(i)},\Sigma_m^{(i)}\right].\end{aligned} \tag{2.49}$$

In this setting, it is possible to prove that $\tilde{\pi}(\mathrm{x}) \to \pi(\mathrm{x})$ as $N \to \infty$ almost surely if $\mu_m^{(i)} \to \mathrm{x}^{(i)}$ and $\Sigma_m^{(i)} \to 0$, by appealing to convergence proofs for mixture-based estimators (see [2], pages 197–199). Also, it is worth noting that this procedure is quite general in the sense that (2.48) could be replaced by a mixture of any convenient parametric distribution.

2.5.2 The Stochastic-Particle-Flow Gaussian Sum Filter

Building upon the results presented in the previous section, SPF-GS uses samples to propagate local Gaussian components, which can together provide an accurate approximation to the posterior probability density of the form (2.48). More specifically, given a new measurement and a set of samples and moments $\{\mathrm{x}_{k-1}^{(i)}, \mu_{m,k-1}^{(i)}, \Sigma_{m,k-1}^{(i)}\}_{i\in[1..N]}$, the filtering procedure consists of integrating the SDE (2.25) for each particle $\mathrm{x}^{(i)}(\lambda)$ and propagating its associated moments $(\mu_m^{(i)}(\lambda), \Sigma_m^{(i)}(\lambda))$ through the interval $0 < \lambda \leq T$, which corresponds to the interval $t_{k-1} < t \leq t_k$. The integration process is performed until one achieves the posterior set of samples and parameters $\{\mathrm{x}_k^{(i)}, \mu_{m,k}^{(i)}, \Sigma_{m,k}^{(i)}\}_{i\in[1..N]} \coloneqq \{\mathrm{x}^{(i)}(T), \mu_m^{(i)}(T), \Sigma_m^{(i)}(T)\}_{i\in[1..N]}$, where $\mathrm{x}^{(i)}(0) = \mathrm{x}_{k-1}^{(i)}$, so that

$$\frac{1}{2}\int_{\mathcal{X}} \left| \sum_{i=1}^{N} \frac{1}{N} \mathcal{N}(\mathrm{x}; \mu_m^{(i)}(T), \Sigma_m^{(i)}(T)) - \pi(\mathrm{x}) \right| d\mathrm{x} \leq \varepsilon',$$

under a desired precision level ε' as $N \to \infty$. In practice, the integration of the stochastic particle flow (2.25) over $0 < \lambda \leq T$ involves multiple intermediate sampling steps that evolve the samples $\mathrm{x}^{(i)}(\lambda)$ to populate the local regions of the state space where the target distribution will be described. At the same time, the mixture components are propagated to define the local approximations and thereby the global approximation to the filtering density. A fundamental aspect of the procedure proposed herein is that, for each filtering step, the intermediate moments $(\mu_m^{(i)}(\lambda), \Sigma_m^{(i)}(\lambda))$ are initialized as departing from the corresponding samples obtained from the previous step, i.e., $\mu_m^{(i)}(0) \coloneqq \mathrm{x}_{k-1}^{(i)}$ and $\Sigma_m^{(i)}(0) \coloneqq 0_{d_x \times d_x}$, and evolved onto the local posterior moments, $\mu_m^{(i)}(T)$ and $\Sigma_m^{(i)}(T)$. In this setting, each component of the filtering mixture is associated, via Fökker-Planck equation (Langevin dynamics), with the mapping

$$\frac{1}{N}\delta(\mathrm{x}_{k-1} - \mathrm{x}_{k-1}^{(i)}) \mapsto \frac{1}{N}\mathcal{N}(\mathrm{x}_k; \mu_{m,k}^{(i)}, \Sigma_{m,k}^{(i)}). \tag{2.50}$$

It is also important to highlight that we perceive the most informative approximation of the posterior from the previous iteration as provided by the set of moments from the previous iteration, $\{\mu_{m,k-1}^{(i)}, \Sigma_{m,k-1}^{(i)}\}_{i\in[1..N]}$, and not by the samples. We therefore use the previous iteration's mixture to define the invariant target measure but samples from this mixture to integrate the SDE. In our experience, this setting is beneficial because: it avoids the bias that would result from propagating a mixture-only approximation from one filtering iteration to the next; it dismisses the need to explicitly compute mixture weights since the ergodic Markov chain that carries out (2.50) is known to converge to the invariant measure no matter where it starts [137].

More specifically, in the Langevin diffusion setting presented in Section 2.4, the mixture measure forms the basis of approximating the target log-density. Each mixture component from the previous filtering step enables a local approximation of the prior density as

$$\tilde{p}^{(i)}\left(\mathrm{x}_k' | \mathrm{y}_{1:k-1}\right) = \int_{\mathbb{R}^{d_x}} p_t(\mathrm{x}_k' | \mathrm{x}_{k-1}) \mathcal{N}(\mathrm{x}_{k-1}; \mu_{m,k-1}^{(i)}, \Sigma_{m,k-1}^{(i)}) d\mathrm{x}_{k-1}, \tag{2.51}$$

where $p_t\left(\mathrm{x}_k' | \mathrm{x}_{k-1}\right)$ is the state-process transition kernel, and the resulting prior density is locally approximated as a Gaussian (e.g., via the Unscented Transform). Thus, provided a

known likelihood function, $p\left(\mathrm{y}_k|\mathrm{x}\right)$, one Langevin transition kernel is computed per sample based on $\nabla_{\mathrm{x}} \log \tilde{\pi}^{(i)}(\mathrm{x}) = \nabla_{\mathrm{x}} \log p\left(\mathrm{y}_k|\mathrm{x}\right) + \nabla_{\mathrm{x}} \log \tilde{p}^{(i)}\left(\mathrm{x}|\mathrm{y}_{1:k-1}\right)$.

We propose an approximate method to propagate the moments, $\mu_m(\lambda)$ and $\Sigma_m(\lambda)$, by linearizing the flow locally, in the neighborhood of a probability mass located at x_l. in Section 2.9.4 we provide an argument (which we regard as useful, if not rigorous) to justify why this local flow approximation should produce acceptable errors on the propagated moments. The procedure produces a negligible error for a small state displacement given a small increment of pseudo-time, $\Delta\lambda$, so that the stochastic particle flow (2.25) can be approximated within the region $\|\mathrm{x} - \mathrm{x}_l\| < \zeta$, for a sufficiently small $\zeta \in \mathbb{R}_+$, as

$$\begin{aligned} d\mathrm{x} &= \frac{1}{2}\mathrm{D}(\lambda)\nabla_{\mathrm{x}} \log \pi\left(\mathrm{x}\right) d\lambda + \mathrm{D}(\lambda)^{1/2} d\mathrm{w}_\lambda, \quad \lambda \in (\lambda_l, \lambda_l + \Delta\lambda],\ \mathrm{x}(\lambda_l) = \mathrm{x}_l; \\ d\mathrm{x} &\approx \left[\mathrm{C}(\mathrm{x}_l, \lambda) \cdot \mathrm{x} + \mathrm{c}(\mathrm{x}_l, \lambda)\right] d\lambda + \mathrm{D}(\lambda)^{1/2} d\mathrm{w}_\lambda. \end{aligned} \tag{2.52}$$

As a consequence of integrating the flow, the corresponding component moments are evolved according to the locally approximated ordinary differential equations (Section 2.9.4)

$$\frac{d\mu_m^{(i)}(\lambda)}{d\lambda} = \mathrm{C}(\mathrm{x}_l^{(i)})\mu_m^{(i)}(\lambda) + \mathrm{c}(\mathrm{x}_l^{(i)}), \tag{2.53}$$

$$\frac{d\Sigma_m^{(i)}(\lambda)}{d\lambda} = \mathrm{C}(\mathrm{x}_l^{(i)})\Sigma_m^{(i)}(\lambda) + \Sigma_m^{(i)}(\lambda)\mathrm{C}^T(\mathrm{x}_l^{(i)}) + \mathrm{D}^{(i)}. \tag{2.54}$$

For nonlinear Gaussian problems, the locally approximated flow implies that

$$\begin{aligned} \mathrm{C}\left(\mathrm{x}_l, \lambda\right) = &-\frac{1}{2}\mathrm{D}(\lambda)\mathrm{P}_{k|k-1}^{-1} \\ &- \frac{1}{2}\mathrm{D}(\lambda)\mathcal{J}_{\mathrm{x}}\left[h(\mathrm{x}_l)\right]^T \mathrm{R}_k^{-1}\mathcal{J}_{\mathrm{x}}\left[h(\mathrm{x}_l)\right], \end{aligned} \tag{2.55}$$

$$\begin{aligned} \mathrm{c}\left(\mathrm{x}_l, \lambda\right) = &\ \frac{1}{2}\mathrm{D}(\lambda)\mathrm{P}_{k|k-1}^{-1} f(\mu_{m,k-1}) \\ &+ \frac{1}{2}\mathrm{D}(\lambda)\mathcal{J}_{\mathrm{x}}\left[h(\mathrm{x}_l)\right]^T \mathrm{R}_k^{-1}\mathcal{J}_{\mathrm{x}}\left[h(\mathrm{x}_l)\right] \cdot \mathrm{x}_l \\ &+ \frac{1}{2}\mathrm{D}(\lambda)\mathcal{J}_{\mathrm{x}}\left[h(\mathrm{x}_l)\right]^T \mathrm{R}_k^{-1}\left(\mathrm{y}_k - h(\mathrm{x}_l)\right), \end{aligned} \tag{2.56}$$

where $\mathcal{J}_{\mathrm{x}}\left[\cdot\right]$ is the Jacobian matrix with respect to the state, $f(\cdot)$ is the state process function, $h(\cdot)$ is the observation function, and where

$$\mathrm{P}_{k|k-1} = \mathbb{E}\left[(\mathrm{x}_{k|k-1} - f(\mu_{m,k-1}))(\mathrm{x}_{k|k-1} - f(\mu_{m,k-1}))^T\right], \tag{2.57}$$

$$\mathrm{R}_k \quad\ = \mathbb{E}\left[(\mathrm{y}_k - h(\mathrm{x}_k))(\mathrm{y}_k - h(\mathrm{x}_k))^T\right], \tag{2.58}$$

are, respectively, the covariance matrix of the prior probability density and the covariance matrix of the observation noise.

Notice that the resulting algorithm is somewhat similar to the Kalman-Bucy filter in that it avoids an explicit discrete-time measurement update. One could also interpret the SPF-GS as a Monte Carlo and continuous-time version of the original Gaussian sum filter [185, 1], albeit modified to explore the Riemannian geometric structure of the probability space. However, in contrast to the Gaussian sum filter (and the particle filter), the structure of the SPF-GS removes the need to explicitly compute mixture weights: it relies on multiple independent (ergodic)

Markov chains that start at the samples to produce equally weighted mixture components, each describing a local version of the underlying geometric structure of the posterior measure.

The SPF-GS has features that are apparently similar to those of the Gaussian sum particle filter [111], however, in reality, these filters rely on distinct fundamental principles that make them very different. The principle of the Gaussian sum particle filter is using importance sampling to estimate the moments of a mixture's components for approximating a target density. In contrast, the SPF-GS evolves a mixture through multiple intermediate steps by exploring the local properties of a stochastic flow in order to translate probability masses from the previous iteration to a mixture on the posterior probability space. The SPF-GS is also very different from that proposed by Terejanu *et al.* [191], which is a Gaussian sum filter analogous to an extended Kalman-Bucy filter but providing an estimate of the predicted mixture weights based on an optimization procedure.

The stochastic particle flow Gaussian-sum filter (SPF-GS) is summarized in *Algorithm 2.1*. The algorithm is expressed in a quite general form, including an accept-reject step that enforces theoretical convergence to the invariant target measure. Our empirical experience indicates that having this step was often not necessary (at least in the cases we have considered). As has already been touched on, if computational efficiency is a primary goal, the SPF-GS should avoid the accept-reject step: calculating the acceptance probability requires double[9] the number of computations of the gradients and inverted Hessian matrices. In addition, using this step requires the explicit evaluation of the kernels and target densities themselves. When the step is removed, there may be the need to deal with outliers, which can significantly affect the estimates because all components in the propagated mixture are treated as equally important. In all numerical examples studied in this chapter, none adopted the Metropolis adjustment. For the high-dimensional examples, the (occasional) outliers are removed online by an empirical test to identify the samples that fall outside a credible inference region.

2.6 Stochastic Particle Flow as a Marginal Particle Filter

In this section, we derive a marginal particle filter whose proposal density is built upon a Gaussian mixture obtained via the stochastic particle flow. The resulting filter is referred to as the stochastic-particle-flow marginal particle filter (SPF-MPF).

2.6.1 Marginal Particle Filtering

In the standard setting, particle filters don't target the marginal filtering distribution $p(\mathrm{x}_k|\mathrm{y}_{1:k})$, a characteristic inherited from the first particle filters, which were designed to be relatively simple to implement. The main problem with the standard particle filters arises because they construct importance densities that target a joint filtering density $p(\mathrm{x}_{0:k}|\mathrm{y}_{1:k})$. A typical particle filter incrementally draws *path samples,* $\{\mathrm{x}_{0:k}^{(i)} \in \mathcal{X}^{k+1}\}_{i\in[1..N]}$, from a joint importance density $q(\mathrm{x}_{0:k}|\mathrm{y}_{1:k})$, and implicitly incorporates the past of the sampled paths $\{\mathrm{x}_{0:k-1}^{(i)} \in \mathcal{X}^{k}\}_{i\in[1..N]}$ when computing (filtered) expectations of interest. Thus, although these algorithms provide

[9]Because of the need to construct both the forward and backward transition kernels.

Algorithm 2.1: Stochastic particle flow - Gaussian sum filter

Initialization:

if *time* $k = 0$ **then**

  Sample $\mathrm{x}_0^{(i)} \sim p(\mathrm{x}_0)$, $\forall i = 1, \ldots, N$

  Set $w_{m,0}^{(i)} \coloneqq N^{-1}$, $\mu_{m,0}^{(i)} \coloneqq \mathbb{E}_{p_0}[\mathrm{x}_0]$, $\Sigma_{m,0}^{(i)} \coloneqq \mathbb{E}_{p_0}\left[(\mathrm{x}_0 - \bar{\mathrm{x}}_0)(\mathrm{x}_0 - \bar{\mathrm{x}}_0)^T\right]$, $\forall i = 1, \ldots, N$

end

Steps:

for *time* $k \geq 1$ **do**

  Compute the time horizon T and step size $\Delta\lambda$ (Section 2.4.6)

  Discretize the interval $0 \leq \lambda \leq T$ into L sub-intervals $\{\lambda_0 = 0, \ldots \lambda_l, \ldots, \lambda_L = T\}$

  Set $\mathrm{x}_{l=0}^{(i)} \coloneqq \mathrm{x}_{k-1}^{(i)}$, $\mu_{l=0}^{(i)} \coloneqq \mathrm{x}_{k-1}^{(i)}$, $\Sigma_{l=0}^{(i)} \coloneqq 0_{d_x \times d_x}$, $\forall i = 1, \ldots, N$

  for $l = 1$ **to** L **do**

    for $i = 1, \ldots, N$ **do**

      Simulate

$$\mathrm{x}_l^{\star(i)} \leftarrow \mathrm{x}_{l-1}^{(i)} + \frac{1}{2}\int_{\lambda_{l-1}}^{\lambda_l} \mathrm{D}(\mathrm{x}_{l-1}^{(i)})\nabla_{\mathrm{x}} \log \tilde{\pi}(\mathrm{x}_{l-1}^{(i)})d\lambda + \int_{\lambda_{l-1}}^{\lambda_l} \mathrm{D}(\mathrm{x}_{l-1}^{(i)})^{1/2} d\mathrm{w}_\lambda$$

      Compute the acceptance probability $\alpha^{(i)} = \min\left[1, \frac{\tilde{\pi}(\mathrm{x}_l^{\star(i)})}{q(\mathrm{x}_l^{\star(i)}|\mathrm{x}_{l-1}^{(i)})} \frac{q(\mathrm{x}_{l-1}^{(i)}|\mathrm{x}_l^{\star(i)})}{\tilde{\pi}(\mathrm{x}_{l-1}^{(i)})}\right]$

      /* We advocate using the Metropolis adjustment step as practically optional, i.e., only for difficult problems. For many Engineering problems the approximation achieved by suppressing the MH step may be enough and will be more computationally efficient. */

      Simulate $u^{(i)} \sim \mathcal{U}(0, 1)$

      if $u^{(i)} \leq \alpha^{(i)}$ **then**

        Set $\mathrm{x}_l^{(i)} \leftarrow \mathrm{x}_l^{\star(i)}$

        Propagate

$$\mu_l^{(i)} \leftarrow \mu_{l-1}^{(i)} + \int_{\lambda_{l-1}}^{\lambda_l} \left[\mathrm{C}(\mathrm{x}_{l-1}^{(i)})\mu_{l-1}^{(i)} + \mathrm{c}(\mathrm{x}_{l-1}^{(i)})\right] d\lambda,$$

$$\Sigma_l^{(i)} \leftarrow \Sigma_{l-1}^{(i)} + \int_{\lambda_{l-1}}^{\lambda_l} \left[\mathrm{C}(\mathrm{x}_{l-1}^{(i)})\Sigma_{l-1}^{(i)} + \Sigma_{l-1}^{(i)}\mathrm{C}^T(\mathrm{x}_{l-1}^{(i)}) + \mathrm{D}(\mathrm{x}_{l-1}^{(i)})\right] d\lambda$$

      else

        Set $\mathrm{x}_l^{(i)} \leftarrow \mathrm{x}_{l-1}^{(i)}$, $\mu_l^{(i)} \leftarrow \mu_{l-1}^{(i)}$, $\Sigma_l^{(i)} \leftarrow \Sigma_{l-1}^{(i)}$

      end

    end

  end

  Set $\mathrm{x}_k^{(i)} \coloneqq \mathrm{x}_{l=L}^{(i)}$, $\mu_{m,k}^{(i)} \coloneqq \mu_{l=L}^{(i)}$, $\Sigma_{m,k}^{(i)} \coloneqq \Sigma_{l=L}^{(i)}$, $\forall i = 1, \ldots, N$

  Output: Approximation of the filtering density as

$$\tilde{p}(\mathrm{x}_k|\mathrm{y}_{1:k}) = \sum_{i=1}^{N} \frac{1}{N} \mathcal{N}(\mathrm{x}_k; \mu_{m,k}^{(i)}, \Sigma_{m,k}^{(i)})$$

end

a simple way to perform measurement update, they perform importance sampling in the joint space along all time steps, i.e., in $\mathcal{X}^{k+1} = \mathcal{X}(0) \times \mathcal{X}(1) \times \cdots \times \mathcal{X}(k)$. The result is precipitation of the *particle degeneracy* phenomenon: the set of paths become increasingly sparse on the joint space $\mathcal{X}^{k+1}$, leading to a quick increase in the weights' variance while most paths have vanishingly small probability. In high-dimensional applications this problem becomes even more pronounced, rendering the standard particle filters to be practically infeasible.

With the mindset of improving this shortcoming in particle filters, Klaas *et al.* [110] proposed the marginal particle filter. The marginal particle filter targets the marginal posterior distribution $p(\mathrm{x}_k|\mathrm{y}_{1:k})$, performing importance sampling on the marginal state space, $\mathcal{X}(k)$, to produce samples with commensurate sparsity over time. The samples are drawn from an importance density of the form

$$q(\mathrm{x}_k|\mathrm{y}_{1:k}) \propto \int_{\mathcal{X}} q(\mathrm{x}_k|\mathrm{x}_{k-1}, \mathrm{y}_k)\, q(\mathrm{x}_{k-1}|\mathrm{y}_{1:k-1})\, d\mathrm{x}_{k-1}, \tag{2.59}$$

to target the posterior density

$$p(\mathrm{x}_k|\mathrm{y}_{1:k}) \propto p(\mathrm{y}_k|\mathrm{x}_k) \int_{\mathcal{X}} p_t(\mathrm{x}_k|\mathrm{x}_{k-1})\, p(\mathrm{x}_{k-1}|\mathrm{y}_{1:k-1})\, d\mathrm{x}_{k-1}, \tag{2.60}$$

with the importance weights

$$w(\mathrm{x}_k) \propto \frac{p(\mathrm{x}_k|\mathrm{y}_{1:k})}{q(\mathrm{x}_k|\mathrm{y}_{1:k})}. \tag{2.61}$$

In practical terms, particles and weights from the previous iteration are used to compose both an approximation of the target density (2.60) and the importance density (2.59), in order to obtain particles and weights for the current iteration. Even though the marginal particle filter is more robust than the standard particle filter against degeneracy, and thereby more suitable to high-dimensional problems in principle, its success is highly dependent on the validity of sequential representations of the target density. Problems may arise in situations where the usual approximation

$$\tilde{p}(\mathrm{x}_k|\mathrm{y}_{1:k}) \propto p(\mathrm{y}_k|\mathrm{x}_k) \sum_{i=1}^{N} w_{k-1}^{(i)} p_t(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)}), \tag{2.62}$$

is prone to relevant statistical or numerical errors, e.g., when the transition density $p_t(\mathrm{x}_k|\mathrm{x}_{k-1})$ describes a Markov process with small variance and the observation y_k lies relatively far from the current set of particles $\{\mathrm{x}_{k-1}^{(i)}, \mathrm{w}_{k-1}^{(i)}\}_{i\in[1..N]}$ on the state space (see the linear, univariate example in Section 2.7). Moreover, owing to the curse of dimensionality, the usual approximation (2.62) is corrupted by a Monte Carlo error that increases geometrically with the number of state dimensions. This may cripple the marginal particle filter in very high-dimensional problems. Because of this limitation, marginal particle filters are likely to perform well only in moderately high-dimensional problems. We illustrate this limitation of marginal particle filters by numerical examples in Section 2.7.

As well covered in [110], there exist several possibilities to choose the marginal importance density (2.59), among which the auxiliary marginal proposal density is particularly interesting because it emulates an optimal importance density in the sense of minimizing the weights'

variance. The marginal optimal (auxiliary) proposal density is usually approximated as

$$\tilde{q}\left(\mathrm{x}_k|\mathrm{y}_{1:k}\right) = \sum_{i=1}^{N} w_{q,k-1}^{(i)} p(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)}, \mathrm{y}_k), \tag{2.63}$$
$$w_{q,k-1}^{(i)} \propto w_{k-1}^{(i)} p(\mathrm{y}_k|\mathrm{x}_{k-1}^{(i)}).$$

It is straightforward to verify that, in the usual setting, the marginal optimal proposal implies that weights never change:

$$\begin{aligned} w_k &\propto \frac{\tilde{p}\left(\mathrm{x}_k|\mathrm{y}_{1:k}\right)}{\tilde{q}\left(\mathrm{x}_k|\mathrm{y}_{1:k}\right)} \propto \frac{p(\mathrm{y}_k|\mathrm{x}_k)\sum_{i=1}^{N} w_{k-1}^{(i)} p_t(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)})}{\sum_{i=1}^{N} w_{q,k-1}^{(i)} p(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)}, \mathrm{y}_k)} \\ &\propto \frac{p(\mathrm{y}_k|\mathrm{x}_k)\sum_{i=1}^{N} w_{k-1}^{(i)} p_t(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)})}{\sum_{i=1}^{N} w_{k-1}^{(i)} p(\mathrm{y}_k|\mathrm{x}_{k-1}^{(i)}) \cdot \frac{p(\mathrm{y}_k|\mathrm{x}_k) p_t(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)})}{p(\mathrm{y}_k|\mathrm{x}_{k-1}^{(i)})}} \\ &= \text{constant}. \end{aligned}$$

This feature is crucial because it endows a particle filter with low variance of weights, which essentially turns into statistical efficiency. This finding motivates the marginal optimal proposal density as the foundation for a marginal particle filter based on the stochastic particle flow. The resulting filter is expected to work well for moderately high-dimensional problems.

2.6.2 Difficulties from a Usual Marginal Importance Density

This section discusses the problems that naturally arise when considering a standard Monte Carlo setting as (2.47) to build a marginal importance density based on the stochastic particle flow. If one regards the proposal distribution as the result of a sequence of L Markov transitions through a discretization of the interval $0 < \lambda \leq T$ onto the sub-intervals $\{\lambda_0 = 0, \ldots \lambda_l, \ldots, \lambda_L = T\}$, where $\mathrm{x}_k \triangleq \mathrm{x}_L$ and $\mathrm{x}_{k-1} \triangleq \mathrm{x}_0$, then the sequence of transitions would provide the importance density

$$\begin{aligned} q\left(\mathrm{x}_k|\mathrm{y}_{1:k}\right) = \int_{\mathcal{X}} \int_{\mathcal{X}} \cdots \int_{\mathcal{X}} & q(\mathrm{x}_L|\mathrm{x}_{L-1}, \mathrm{y}_k) q(\mathrm{x}_{L-1}|\mathrm{x}_{L-2}, \mathrm{y}_k) \ldots \\ & q(\mathrm{x}_1|\mathrm{x}_0, \mathrm{y}_k) q(\mathrm{x}_0|\mathrm{y}_{1:k-1}) d\mathrm{x}_{L-1} d\mathrm{x}_{L-2} \ldots d\mathrm{x}_0 \cdot \end{aligned} \tag{2.64}$$

In order to evaluate this importance density over a set of N particles, incorporating the previous set of samples and importance weights, one would be required to compute

$$\tilde{q}(\mathrm{x}_k^{(i)}|\mathrm{y}_{1:k}) = \sum_{j=1}^{N} w_{k-1}^{(j)} \tilde{q}(\mathrm{x}_k^{(i)}|\mathrm{x}_{k-1}^{(j)}, \mathrm{y}_k), \tag{2.65}$$
$$i = 1, \ldots, N.$$

This implementation depends on the set of conditional kernels $\tilde{q}(\mathrm{x}_k|\mathrm{x}_{k-1}, \mathrm{y}_k) = \tilde{q}(\mathrm{x}_L|\mathrm{x}_0, \mathrm{y}_k)$

that could be achieved in terms of a recursion of the form

$$
\begin{aligned}
w_1^{(i|j)} &\triangleq q(\mathrm{x}_1^{(i)}|\mathrm{x}_0^{(j)}, \mathrm{y}_k), \quad i,\, j = 1, \ldots, N; \\
w_2^{(i|j)} &\triangleq \tilde{q}(\mathrm{x}_2^{(i)}|\mathrm{x}_0^{(j)}, \mathrm{y}_k) = \sum_{n=1}^{N} w_1^{(n|j)} q(\mathrm{x}_2^{(i)}|\mathrm{x}_1^{(n)}, \mathrm{x}_0^{(j)}, \mathrm{y}_k); \\
&\vdots \\
w_l^{(i|j)} &\triangleq \tilde{q}(\mathrm{x}_l^{(i)}|\mathrm{x}_0^{(j)}, \mathrm{y}_k) = \sum_{n=1}^{N} w_{l-1}^{(n|j)} q(\mathrm{x}_l^{(i)}|\mathrm{x}_{l-1}^{(n)}, \mathrm{x}_0^{(j)}, \mathrm{y}_k); \\
&\vdots \\
\tilde{q}(\mathrm{x}_L^{(i)}|\mathrm{x}_0^{(j)}, \mathrm{y}_k) &= \sum_{n=1}^{N} w_{L-1}^{(n|j)} q(\mathrm{x}_L^{(i)}|\mathrm{x}_{L-1}^{(n)}, \mathrm{x}_0^{(j)}, \mathrm{y}_k);
\end{aligned} \tag{2.66}
$$

where $q(\mathrm{x}_l|\mathrm{x}_{l-1}, \mathrm{x}_0, \mathrm{y}_k)$ are the one-step proposal kernels conditioned on the initial state (prior samples), which are directly available from the discretized version of (2.25). Computing the conditional proposal components (2.66) and the marginal proposal (2.65) involves high computational effort, bounded by $(L-1)N^3 + N^2 \sim \mathcal{O}(LN^3)$ evaluations. In addition, the main complication of this realization is due to the mixing properties of (2.65), leading to significant errors built up through the sequence of finite-sample approximations in (2.66) along with the prohibitively high variance of the resulting importance weights (2.61).

While these problems could be tentatively worked around by a judicious choice of a variance reduction method, it is worth looking how the implementation difficulties would turn out to be by evoking a hypothetical "continuity" between sampling steps. It is well known that in the limit $\Delta\lambda \to 0$, the proposal density (2.64) defines a path integral. Based on the concept of path probability density [84] of a Markov process

$$
W_\infty\left[\mathrm{x}\left(\lambda\right)\right]\left[d\mathrm{x}\right] \propto e^{-\int_0^T \left[\frac{1}{2}(\dot{\mathrm{x}}-\mu(\mathrm{x}))^T \mathrm{D}^{-1}(\dot{\mathrm{x}}-\mu(\mathrm{x}))+\frac{1}{2}\nabla_{\mathrm{x}}\cdot\mu(\mathrm{x})\right]d\lambda}, \tag{2.67}
$$

for samples describing continuous paths, the proposal could be written as a functional integral [100] of the form

$$
q_c\left(\mathrm{x}_k|\mathrm{y}_{1:k}\right) \propto \int e^{-\int_0^T \left[\frac{1}{2}(\dot{\mathrm{x}}-\mu(\mathrm{x}))^T \mathrm{D}^{-1}(\dot{\mathrm{x}}-\mu(\mathrm{x}))+\frac{1}{2}\nabla_{\mathrm{x}}\cdot\mu(\mathrm{x})\right]d\lambda}\left[d\mathrm{x}\right], \tag{2.68}
$$

where $[d\mathrm{x}] = d\mathrm{x}_{L-1} \ldots d\mathrm{x}_0$ as $\Delta\lambda \to 0$. Solving path integrals in general is a daunting task, nevertheless, a density of interest could be approximately obtained in terms of a Gaussian mixture, under the assumption of local Gaussianity of probability paths. Within this framework, an ensemble of independently selected Gaussian densities can be analytically integrated to achieve local solutions to (2.68). This fundamental idea is equivalent to what the stochastic particle flow proposes when the filtering solution is formulated as the mixture (2.48).

2.6.3 The Stochastic-Particle-Flow Marginal Particle Filter

In marginal particle filtering, the best importance density one could achieve is the proposal density $q\left(\mathrm{x}_k|\mathrm{y}_{1:k}\right)$ when computed exactly. This density enables inference of the actual posterior pdf, $p\left(\mathrm{x}_k|\mathrm{y}_{1:k}\right)$. Composing the marginal optimal proposal requires computing $p(\mathrm{y}_k|\mathrm{x}_{k-1}^{(i)})$

exactly, which is not possible in general. In addition, the same scenarios that produce considerable errors in computing the empirical target, $\tilde{\pi}\left(\mathrm{x}_k\right) = \tilde{p}\left(\mathrm{x}_k|\mathrm{y}_{1:k}\right)$, according to (2.62), will also affect evaluation of the proposal $\tilde{q}\left(\mathrm{x}_k|\mathrm{y}_{1:k}\right)$, computed by (2.63), as illustrated by the first example in Section 2.7. In these cases, one can benefit from the inherent characteristics of the stochastic particle flow to construct a proposal density with better regularity properties by doing

$$\begin{aligned}
\tilde{q}\left(\mathrm{x}_k|\mathrm{y}_{1:k}\right) &= \sum_{i=1}^{N} w_{k-1}^{(i)} p(\mathrm{y}_k|\mathrm{x}_k) p_t(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)}) \\
&= \sum_{i=1}^{N} w_{k-1}^{(i)} \frac{p_t(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)})}{\tilde{p}^{(i)}(\mathrm{x}_k|\mathrm{y}_{1:k-1})} p(\mathrm{y}_k|\mathrm{x}_k) \tilde{p}^{(i)}(\mathrm{x}_k|\mathrm{y}_{1:k-1}) \\
&\propto \sum_{i=1}^{N} w_{k-1}^{(i)} \frac{p_t(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)})}{\tilde{p}^{(i)}(\mathrm{x}_k|\mathrm{y}_{1:k-1})} w_{m,k}^{(i)} \mathcal{N}(\mathrm{x}_k; \mu_{m,k}^{(i)}, \Sigma_{m,k}^{(i)}), \\
\tilde{q}\left(\mathrm{x}_k|\mathrm{y}_{1:k}\right) &= \sum_{i=1}^{N} w_q^{(i)}(\mathrm{x}_k) \mathcal{N}(\mathrm{x}_k; \mu_{m,k}^{(i)}, \Sigma_{m,k}^{(i)}); \qquad (2.69)
\end{aligned}$$

where

$$\bar{w}_q^{(i)}(\mathrm{x}_k) = w_{k-1}^{(i)} w_{m,k}^{(i)} \frac{p_t(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)})}{\tilde{p}^{(i)}(\mathrm{x}_k|\mathrm{y}_{1:k-1})}, \qquad (2.70)$$

$$w_q^{(i)}(\mathrm{x}_k) = \frac{\bar{w}_q^{(i)}(\mathrm{x}_k)}{\sum_{i=1}^{N} \bar{w}_q^{(i)}(\mathrm{x}_k)}; \qquad (2.71)$$

and $\tilde{p}^{(i)}(\mathrm{x}_k|\mathrm{y}_{1:k-1})$ is a per-sample, local prior density. For a known Markov transition density $p_t\left(\mathrm{x}'_k|\mathrm{x}_{k-1}\right)$, we recall the local prior density as given by

$$\tilde{p}^{(i)}\left(\mathrm{x}'_k|\mathrm{y}_{1:k-1}\right) = \int_{\mathcal{X}} p_t(\mathrm{x}'_k|\mathrm{x}_{k-1}) \mathcal{N}(\mathrm{x}_{k-1}; \mu_{m,k-1}^{(i)}, \Sigma_{m,k-1}^{(i)}) d\mathrm{x}_{k-1}. \qquad (2.72)$$

As in the classical Gaussian-sum setting, the mixture weights $\{w_{m,k}^{(i)}\}_{i\in[1..N]}$ are given by (see [2], pages 214 and 215)

$$w_{m,k}^{(i)} \propto \frac{1}{N} \int_{\mathcal{X}} p\left(\mathrm{y}_k|\mathrm{x}'_k\right) \tilde{p}^{(i)}\left(\mathrm{x}'_k|\mathrm{y}_{1:k-1}\right) d\mathrm{x}'_k, \qquad (2.73)$$

where the proportionality to N^{-1} holds because the stochastic particle flow generates equally weighted mixture components. The mixture weights $\{w_{m,k}^{(i)}\}$ generated by this method are only applicable in the context of the proposal (2.69), and shall be re-evaluated in the same way whenever a new instance of the marginal proposal is constructed. It is relevant to make clear the distinction $\mathrm{x}'_k \neq \mathrm{x}_k$ in the expressions (2.73) and (2.72), bearing in mind that x'_k corresponds to the state that the flow would reach when considering only the prior density as the target $\pi_{\text{prior}}(\mathrm{x}') = p_x(\mathrm{x}') \triangleq p(\mathrm{x}'|\mathrm{y}_{1:k-1})$. We note that the involved integrals may not be tractable in general and may require approximation either by a Gaussian representation of the likelihood, or adequate quadrature rules (e.g., Gauss-Hermite).

This formulation evokes the stochastic particle flow to promote an accurate approximation to the marginal optimal proposal density. Given a set of samples and parameters from a previous

filtering iteration, $\{\mathrm{x}_{k-1}^{(i)}, w_{k-1}^{(i)}, \mu_{m,k-1}^{(i)}, \Sigma_{m,k-1}^{(i)}\}_{i\in[1..N]}$, where $w_{k-1}^{(i)}$ are importance weights, the algorithm integrates the SDE (2.25) for each sample and propagates the associated parameters through the interval $0 < \lambda \leq T$. As a result, the procedure acquires posterior samples and parameters, $\{\mathrm{x}_{k}^{(i)}, w_{k}^{(i)}, \mu_{m,k}^{(i)}, \Sigma_{m,k}^{(i)}\}_{i\in[1..N]}$, which are used to evaluate the marginal proposal (2.69) and enable filtering as by a marginal particle filter. The moments of the mixture's components are evolved in accordance with (2.53) and (2.54), and the importance weights are updated by

$$w_k(\mathrm{x}_k) \propto \frac{\sum_{j=1}^{N} w_{k-1}^{(j)} p(\mathrm{y}_k|\mathrm{x}_k) p_t(\mathrm{x}_k|\mathrm{x}_{k-1}^{(j)})}{\sum_{j=1}^{N} w_q^{(j)}(\mathrm{x}_k)\mathcal{N}(\mathrm{x}_k; \mu_{m,k}^{(j)}, \Sigma_{m,k}^{(j)})}. \tag{2.74}$$

The resulting filter, called stochastic-particle-flow marginal particle filter (SPF-MPF), is summarized in *Algorithm 2.2*. It is worth noting that a simpler alternative to (2.69) could be chosen by considering

$$\tilde{q}\left(\mathrm{x}_k|\mathrm{y}_{1:k}\right) = \sum_{i=1}^{N} \frac{1}{N} \mathcal{N}(\mathrm{x}_k; \mu_{m,k}^{(i)}, \Sigma_{m,k}^{(i)}), \tag{2.75}$$

however, in that case, the importance density would not be affected by the same errors as the empirical target, $\tilde{\pi}\left(\mathrm{x}_k\right) = \tilde{p}\left(\mathrm{x}_k|\mathrm{y}_{1:k}\right)$, as computed by (2.62), because each component in (2.75) targets a local instance of the posterior density itself. As a result, even though the importance density could approximate the true posterior density accurately, it would not directly approach the target density. In situations where the empirical target density cannot represent the true posterior density as well as a mixture of the form (2.75), the SPF-MPF with such a proposal would fail because of the mismatch originated from distinctions in the approximation methods. As a consequence, the importance weights would have infeasibly high variance. The described issue is equivalent to treat errors in the standard Monte Carlo measure (2.47) as comparable to errors in the mixture measure (2.49), which is not true except for rare cases. This scenario is well illustrated by two examples in Section 2.7.

2.7 Examples

In this section, we present some illustrative toy examples and experimental results for three instructive applications in the multi-sensor multi-target tracking context: a multi-sensor bearing-only problem, a convoy tracking problem, and an inference on a large spatial sensor network as presented by Septier & Peters [175].

In the experimental results for the bearing-only and convoy tracking examples, we compared the SPF-GS against standard target trackers and extensions of two of the most effective particle flows, namely, the Gaussian particle flow (GPF) and the scaled-drift particle flow (SDPF). The GPF was first called *exact particle flow* in [53] and the SDPF was first called *non-zero diffusion particle flow* in [52]. Actually, this latter is a particle flow with the drift scaled by a diffusion coefficient but the filter itself is not a diffusion.

It is important to mention that, in order to work properly, both the Gaussian particle flow and the scaled-drift particle flow are implemented with the aid of a companion filter such that the state covariance matrix can be correctly estimated. Implementation details have been presented by Choi *et al.* [30] and Ding & Coates [63], who advocate using the EKF (or UKF) as

Algorithm 2.2: Stochastic particle flow - marginal particle filter

Initialization:

if *time* $k = 0$ **then**

  Sample $\mathrm{x}_0^{(i)} \sim p(\mathrm{x}_0)$ and set $w_0^{(i)} \coloneqq N^{-1}$, $\forall i = 1, \ldots, N$

  Set $w_{m,0}^{(i)} \coloneqq N^{-1}$, $\mu_{m,0}^{(i)} \coloneqq \mathbb{E}_{p_0}[\mathrm{x}_0]$, $\Sigma_{m,0}^{(i)} \coloneqq \mathbb{E}_{p_0}\left[(\mathrm{x}_0 - \bar{\mathrm{x}}_0)(\mathrm{x}_0 - \bar{\mathrm{x}}_0)^T\right]$, $\forall i = 1, \ldots, N$

end

Steps:

for *time* $k \geq 1$ **do**

  Compute the time horizon T and step size $\Delta\lambda$ (Section 2.4.6)

  Discretize the interval $0 \leq \lambda \leq T$ into L sub-intervals $\{\lambda_0 = 0, \ldots \lambda_l, \ldots, \lambda_L = T\}$

  Set $\mathrm{x}_{l=0}^{(i)} \coloneqq \mathrm{x}_{k-1}^{(i)}$, $\mu_{l=0}^{(i)} \coloneqq \mathrm{x}_{k-1}^{(i)}$, $\Sigma_{l=0}^{(i)} \coloneqq 0_{d_x \times d_x}$, $\forall i = 1, \ldots, N$

  for $l = 1$ **to** L **do**

    for $i = 1, \ldots, N$ **do**

      Simulate

$$\mathrm{x}_l^{(i)} \leftarrow \mathrm{x}_{l-1}^{(i)} + \frac{1}{2}\int_{\lambda_{l-1}}^{\lambda_l} \mathrm{D}(\mathrm{x}_{l-1}^{(i)}) \nabla_{\mathrm{x}} \log \tilde{\pi}(\mathrm{x}_{l-1}^{(i)}) d\lambda + \int_{\lambda_{l-1}}^{\lambda_l} \mathrm{D}(\mathrm{x}_{l-1}^{(i)})^{1/2} d\mathrm{w}_\lambda$$

      Propagate

$$\mu_l^{(i)} \leftarrow \mu_{l-1}^{(i)} + \int_{\lambda_{l-1}}^{\lambda_l} \left[\mathrm{C}(\mathrm{x}_{l-1}^{(i)})\mu_{l-1}^{(i)} + \mathrm{c}(\mathrm{x}_{l-1}^{(i)})\right] d\lambda,$$

$$\Sigma_l^{(i)} \leftarrow \Sigma_{l-1}^{(i)} + \int_{\lambda_{l-1}}^{\lambda_l} \left[\mathrm{C}(\mathrm{x}_{l-1}^{(i)})\Sigma_{l-1}^{(i)} + \Sigma_{l-1}^{(i)}\mathrm{C}^T(\mathrm{x}_{l-1}^{(i)}) + \mathrm{D}(\mathrm{x}_{l-1}^{(i)})\right] d\lambda$$

    end

  end

  Set $\mathrm{x}_k^{(i)} \coloneqq \mathrm{x}_{l=L}^{(i)}$, $\mu_{m,k}^{(i)} \coloneqq \mu_{l=L}^{(i)}$, $\Sigma_{m,k}^{(i)} \coloneqq \Sigma_{l=L}^{(i)}$, $\forall i = 1, \ldots, N$

  Compute the normalized proposal weights, $\forall i = 1, \ldots, N$, by

$$w_{m,k}^{(i)} \propto \frac{1}{N}\int_{\mathcal{X}} p\left(\mathrm{y}_k | \mathrm{x}'_k\right) \tilde{p}^{(i)}\left(\mathrm{x}'_k | \mathrm{y}_{1:k-1}\right) d\mathrm{x}'_k,$$

$$w_q^{(i)} \propto w_{k-1}^{(i)} w_{m,k}^{(i)} \frac{p_t(\mathrm{x}_k^{(i)} | \mathrm{x}_{k-1}^{(i)})}{\tilde{p}^{(i)}(\mathrm{x}_k^{(i)} | \mathrm{y}_{1:k-1})}$$

  Compute the normalized importance weights, $\forall i = 1, \ldots, N$, by

$$w_k^{(i)} \propto \frac{\sum_{j=1}^N w_{k-1}^{(j)} p(\mathrm{y}_k | \mathrm{x}_k^{(i)}) p_t(\mathrm{x}_k^{(i)} | \mathrm{x}_{k-1}^{(j)})}{\sum_{j=1}^N w_q^{(j)} \mathcal{N}(\mathrm{x}_k^{(i)}; \mu_{m,k}^{(j)}, \Sigma_{m,k}^{(j)})}$$

  if $ESS_k < 0.5N$ **then** resample: $\{\mathrm{x}_k^{(i)}, N^{-1}\} \leftarrow \{\mathrm{x}_k^{(i)}, w_k^{(i)}\}$

  Output: Approximation of the filtering distribution by the empirical measure

$$\tilde{p}(\mathrm{x}_k | \mathrm{y}_{1:k}) = \sum_{i=1}^N w_k^{(i)} \delta(\mathrm{x}_k - \mathrm{x}_k^{(i)})$$

end

a companion filter to estimate the associated covariance matrices. Another option is to shrink the empirical covariance and apply Tikhonov regularization [109]. In contrast, the stochastic particle flow does not require any auxiliary technique to estimate the second order moment, relying solely on its mixture measure. In the toy examples, a companion filter was not necessary for the original particle flows since a single filtering cycle has been analyzed. In the multi-sensor and multi-target examples, we adopted baseline filters, which are the most structurally similar to the EKF, as companion filters for the particle flows (GPF, SDPF).

In the example of the large spatial sensor network, we compared the SPF-GS, a particle filter (Sequential Importance Resampling - SIR), a block particle filter (block SIR), and two of the best sequential MCMC filters [80, 175]: the Sequential manifold Metropolis-Adjusted Algorithm (SmMALA) and the Sequential manifold Hamiltonian Monte Carlo (SmHMC). The block particle filter partitions the state space into separate subspaces of smaller dimensions and run a particle filter on each subspace [160].

2.7.1 Toy Examples

The toy examples are based on Gaussian processes chosen to demonstrate the properties of the stochastic particle flow, summarized as

- Univariate
  - linear,
  - quadratic,
  - cubic;
- Bivariate
  - multimodal, linear,
  - nonlinear (banana-shaped pdf).

In all cases, we analyze the filters for a single filtering cycle. Generally, we describe the state process, the observation process and the initial distribution for these examples as

$$\mathrm{x}_k = f(\mathrm{x}_{k-1}) + \mathrm{u}_k,\ \mathrm{u}_k \sim \mathcal{N}(\mathrm{u}_k; 0, \mathrm{Q}_k), \tag{2.76}$$

$$\mathrm{y}_k = h(\mathrm{x}_k\ \ ) + \mathrm{v}_k,\ \mathrm{v}_k \sim \mathcal{N}(\mathrm{v}_k; 0, \mathrm{R}_k), \tag{2.77}$$

$$p_0(\mathrm{x}_{k-1}) = \mathcal{N}(\mathrm{x}_{k-1}; \bar{\mathrm{x}}_{k-1}, \mathrm{P}_{k-1}). \tag{2.78}$$

We consider four different types of particle filters based on the marginal importance density

$$\tilde{q}\left(\mathrm{x}_k|\mathrm{y}_{1:k}\right) = \sum_{i=1}^{N} w_{k-1}^{(i)} q(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)}, \mathrm{y}_k),$$

where

- for the marginal bootstrap particle filter (MBPF), the proposal's components are set as the Markov transition kernel: $q(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)}, \mathrm{y}_k) = p_t(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)})$;
- for the marginal EKF-based particle filter (MEPF), the proposal's components are computed by the EKF: $q(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)}, \mathrm{y}_k) = p_{\mathrm{EKF}}(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)}, \mathrm{y}_k)$;

- for the marginal UKF-based particle filter (MUPF), the proposal's components are computed by the UKF: $q(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)}, \mathrm{y}_k) = p_{\mathrm{UKF}}(\mathrm{x}_k|\mathrm{x}_{k-1}^{(i)}, \mathrm{y}_k)$; and
- for the marginal auxiliary particle filter (MAPF) [110], the importance density is given by (2.63).

When comparing probability densities furnished by different filters, we include the empirical marginal target, $\tilde{\pi}\left(\mathrm{x}_k\right) = \tilde{p}\left(\mathrm{x}_k|\mathrm{y}_{1:k}\right)$, evaluated according to (2.62) for samples obtained by the stochastic particle flow. For all filters, when applicable, we calculate the average of the effective sample size

$$\mathrm{ESS}_k = \left(\sum_{i=1}^{N} w_k^{(i)\,2}\right)^{-1} \tag{2.79}$$

over 100 Monte Carlo runs, for 1000 particles. For all marginal proposal densities, we analyze their similarity to the true posterior probability density by averaging their empirical Jensen-Shannon divergence (JSD) with respect to the true posterior, which is obtained to high numerical precision. The Jensen-Shannon divergence is defined as

$$\begin{aligned}\mathrm{JSD}\left(\mathcal{P} \parallel \mathcal{Q}\right) = &\frac{1}{2}\mathrm{D}_{\mathrm{KL}}(\mathcal{P} \parallel \left(\mathcal{P} + \mathcal{Q}\right)/2) \\ &+ \frac{1}{2}\mathrm{D}_{\mathrm{KL}}(\mathcal{Q} \parallel \left(\mathcal{P} + \mathcal{Q}\right)/2),\end{aligned} \tag{2.80}$$

where the Kullback–Leibler divergence, $\mathrm{D}_{\mathrm{KL}}(\cdot \parallel \cdot)$, is computed using the base-2 logarithm such that the Jensen-Shannon divergence is bounded as $0 \leq \mathrm{JSD}\left(\mathcal{P} \parallel \mathcal{Q}\right) \leq 1$. The Jensen-Shannon divergence is symmetric and equals zero when the compared densities are equal. In the bivariate examples, we also consider the original particle flow methods, the Gaussian particle flow (GPF) and scaled-drift particle flow (SDPF), for which the Jensen-Shannon divergence with respect to the true posterior is evaluated based on empirical densities constructed by (bidimensional) histograms of samples.

Linear, Univariate Model

The simplest example is a linear, univariate model, with parameters set as in the table below.

Parameters for the linear, univariate model	
Initial distribution	$\bar{\mathrm{x}}_{k-1} = 0$, $\mathrm{P}_{k-1} = 20$
Markov transition pdf	$f(\mathrm{x}_{k-1}) = \mathrm{x}_{k-1}$, $\mathrm{Q}_k = 5$
Likelihood function	$h(\mathrm{x}_k) = \mathrm{x}_k$, $\mathrm{R}_k = 10$
Observation	$\mathrm{y}_k = 30$

Table 2.1: Parameters for the linear univariate model

Although very simple, this example was proposed to demonstrate a scenario where the empirical marginal target, $\tilde{\pi}\left(\mathrm{x}_k\right) = \tilde{p}\left(\mathrm{x}_k|\mathrm{y}_{1:k}\right)$, is prone to relevant statistical and numerical errors. This is done by setting a situation where the transition kernel describes a Markov process with small variance and the observation lies relatively far from the initial distribution. In this scenario, statistical inefficiency emerges because the observation provides little information in the space region where probability masses are more densely distributed by the state process. Not

incidentally, this is also the main source of degeneracy in standard particle filters. Additionally, there may exist round-off errors when evaluating the empirical marginal target owing to samples being located relatively far from the posterior mean, several standard deviations apart, in the tail of each proposal component.

As depicted in Figure 2.1, the importance density proposed by the SPF-MPF (red x's) is successful at aiming the empirical marginal target (blue circles), generating a high effective sample size. However, since the empirical target constitutes a poor approximation to the true posterior pdf (black line), importance sampling clearly fails and the SPF-MPF leads to a solution excessively biased. In contrast, the direct filtering density generated by the SPF-GS approximates the true posterior pdf accurately, generating a satisfactory solution. These findings are quantified by the Jensen-Shannon divergences averaged over 100 Monte Carlo runs and presented in Table 2.4. Table 2.4 shows a negligible divergence between the density filtered by the SPF-GS and the true posterior whereas the divergences computed for the target density and for the proposal density constructed by the SPF-MPF are significant.

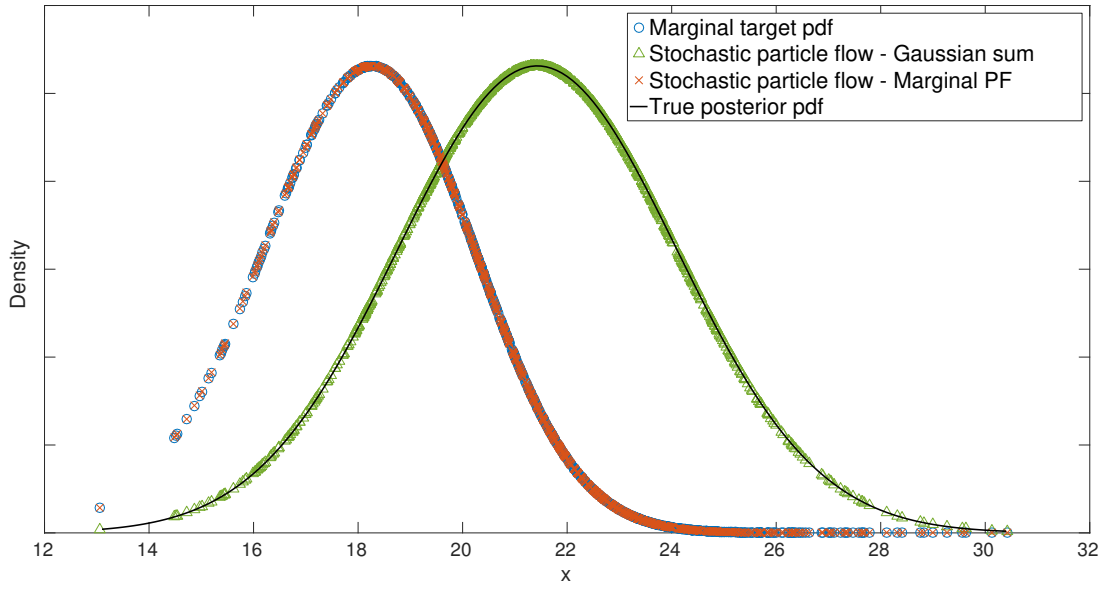


Figure 2.1: Densities generated by the SPF for the linear, univariate example

Quadratic, Univariate Model

The quadratic, univariate model was tested with parameters set as shown in the following table. This model is interesting because nonlinearity of the observation process leads to bimodality of the filtered density.

Parameters for the quadratic, univariate model	
Initial distribution	$\bar{\mathrm{x}}_{k-1} = 0$, $\mathrm{P}_{k-1} = 20$
Markov transition pdf	$f(\mathrm{x}_{k-1}) = \mathrm{x}_{k-1}$, $\mathrm{Q}_k = 20$
Likelihood function	$h(\mathrm{x}_k) = \mathrm{x}_k^2/20$, $\mathrm{R}_k = 50$
Observation	$\mathrm{y}_k = 30$

Table 2.2: Parameters for the quadratic univariate model

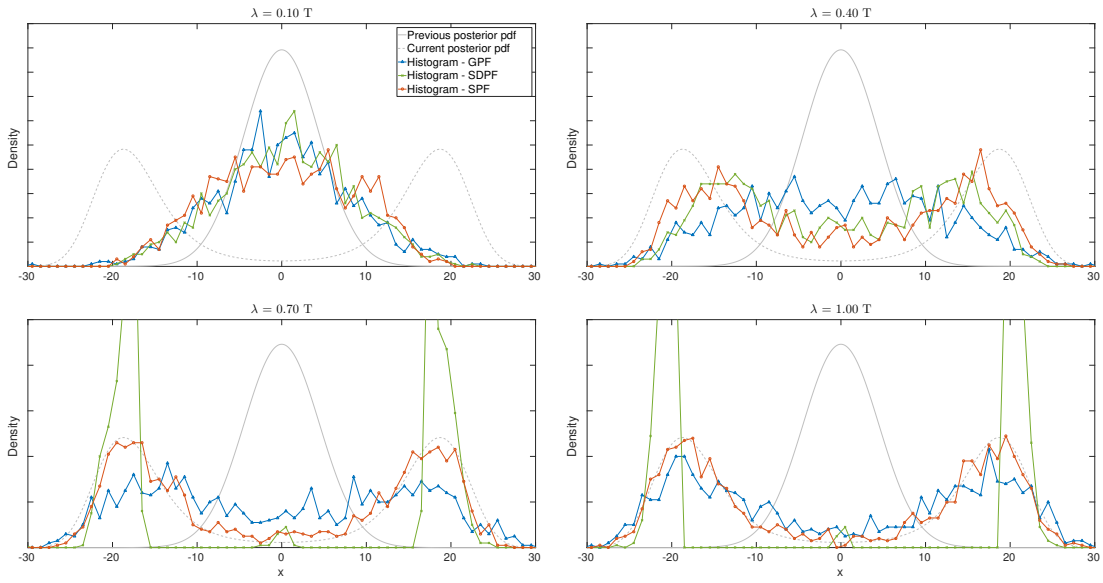


Figure 2.2: Sequence of histograms for the quadratic, univariate example

This nonlinear example was set to be favorable for marginal importance sampling such that it would be possible to compare different marginal particle filters against the SPF-MPF. The original particle flows, GPF and SDPF, are compared to the SPF-MPF as well. The quantified performances for this quadratic univariate model are shown in Table 2.4.

Firstly, we compare the sequence of histograms achieved when propagating samples by the GPF, by the SDPF and by the SPF-GS. As it can be seen in Figure 2.2, for this example, the stochastic particle flow provides the best distribution of particles to approximate the posterior density, denoting a higher level of accuracy and regularity of the flow formulated as a diffusion.

Regarding the marginal importance densities illustrated in Figure 2.3, we observe a high degree of similarity of the SPF-MPF proposal density to the marginal target density. In the same manner, the filtering density achieved by the SPF-GS accurately approximates the true posterior density, as evidenced in Table 2.4. In Figure 2.4 we can see in detail the proximity of the SPF-MPF proposal density to both the marginal target density and the true posterior density, along with some of the proposal mixture components. The density proposed by the marginal (optimal) auxiliary particle filter (MAPF) is also very similar to the marginal target, providing an accurate solution, whereas all other filters propose densities less effective for this example. These observations are quantitatively captured by the performance data summarized in Table 2.4.

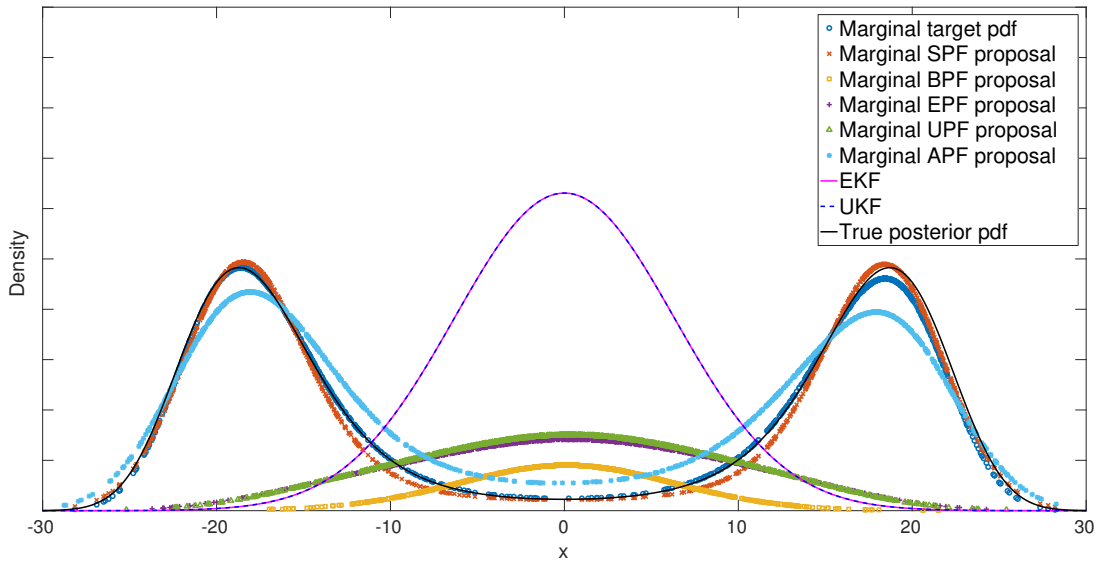


Figure 2.3: Comparison of proposal densities for the quadratic, univariate example

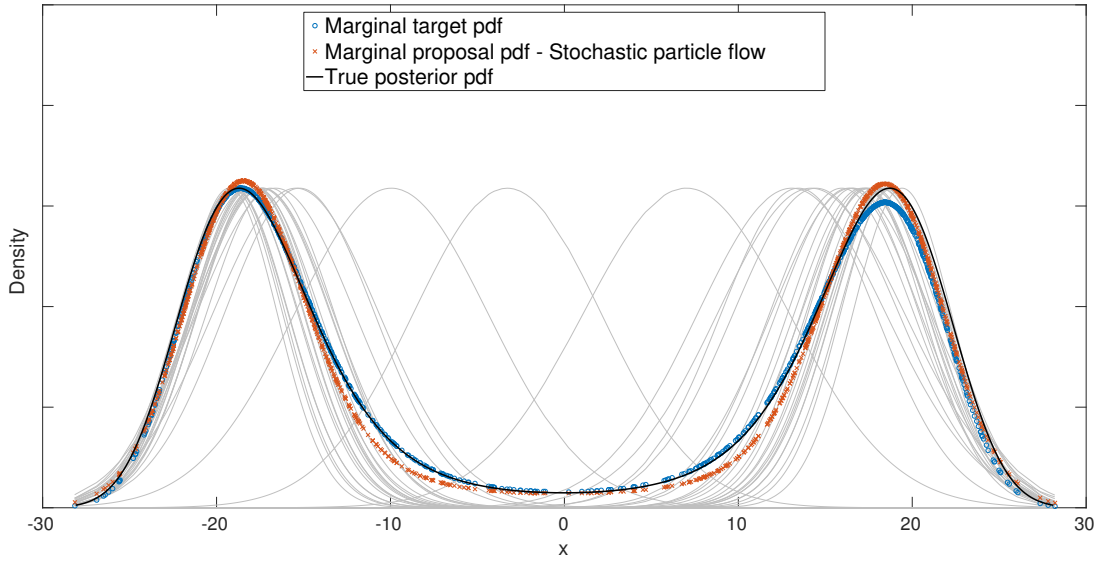


Figure 2.4: Marginal proposal density based on the stochastic particle flow

Cubic, Univariate Model

The cubic, univariate model was tested with parameters set as shown in the following table.

Parameters for the cubic, univariate model	
Initial distribution	$\bar{\mathrm{x}}_{k-1} = 0$, $\mathrm{P}_{k-1} = 20$
Markov transition pdf	$f(\mathrm{x}_{k-1}) = \mathrm{x}_{k-1}$, $\mathrm{Q}_k = 20$
Likelihood function	$h(\mathrm{x}_k) = \mathrm{x}_k^3/120$, $\mathrm{R}_k = 50$
Observation	$\mathrm{y}_k = 20$

Table 2.3: Parameters for the cubic univariate model

This nonlinear example was also set to be favorable for marginal importance sampling, i.e., avoiding the scenario described in the first toy example where importance sampling fails. By comparing the resulting histograms achieved when propagating samples by the GPF, by the SDPF and by the stochastic particle flow, it is remarkable in Figure 2.5 that the stochastic particle flow provides a fairly superior distribution of particles to approximate the posterior density. This superiority is incorporated in the importance density proposed by the SPF-MPF as can be seen in Figure 2.6. The importance density proposed by the marginal auxiliary particle filter (MAPF) also provides an accurate solution to the filtering problem but it is slightly less effective than the SPF-MPF. All the other marginal particle filters present less effective solutions. The comparison of all filters for this example is quantified in Table 2.4.

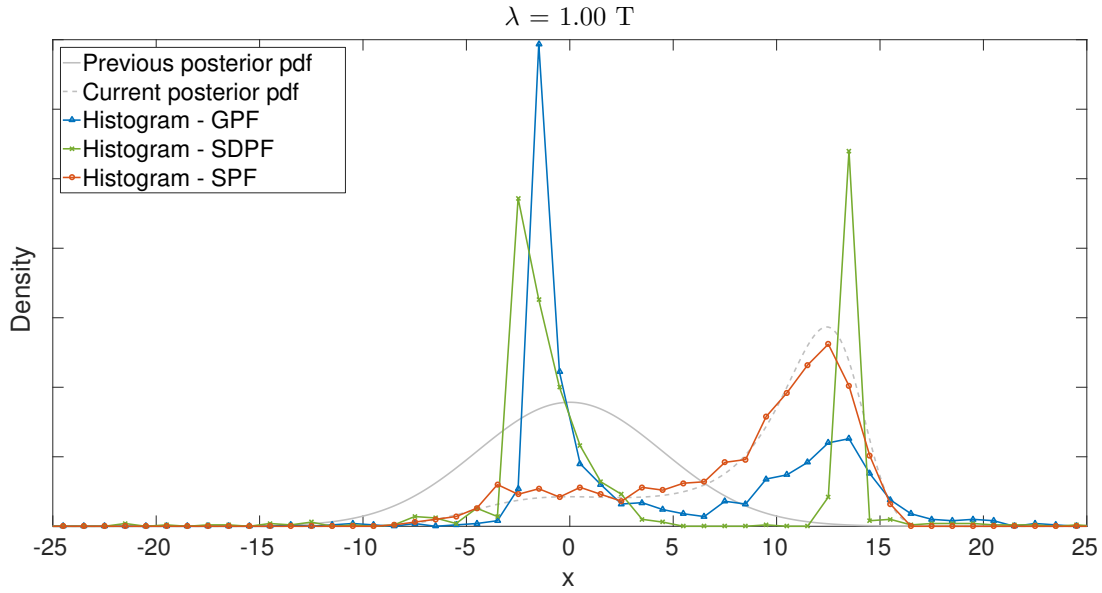


Figure 2.5: Resulting histograms of particles for the cubic, univariate example

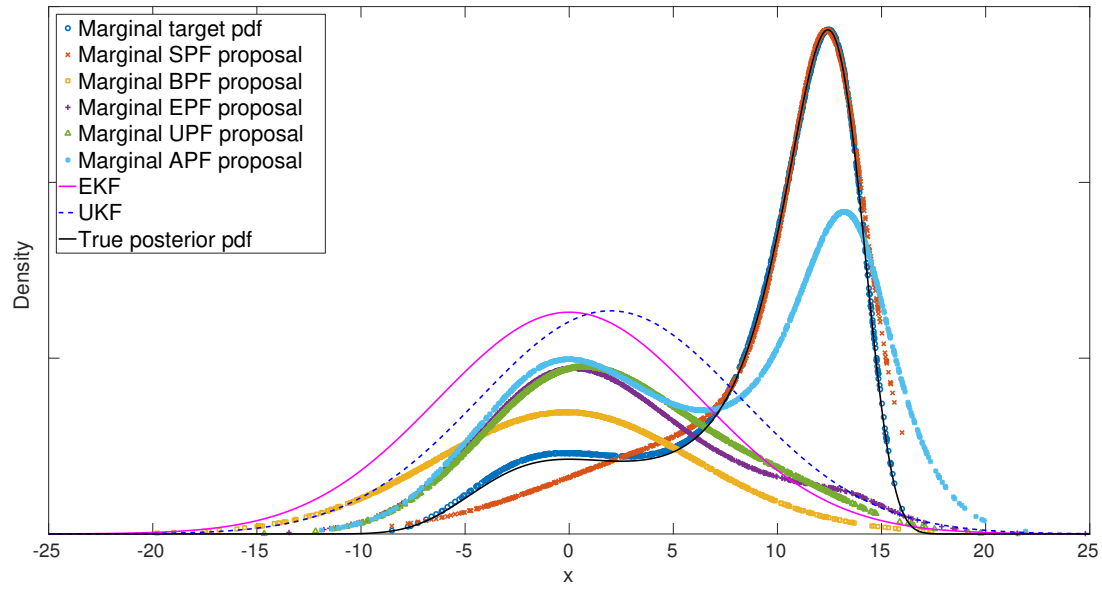


Figure 2.6: Comparison of proposal densities for the cubic, univariate example

Density	*Linear*		*Quadratic*		*Cubic*	
	JSD_{avg}	ESS_{avg}	JSD_{avg}	ESS_{avg}	JSD_{avg}	ESS_{avg}
Marginal target	0.1572	-	0.0028	-	0.0001	-
SPF-GS	0.0000	-	0.0013	-	0.0165	-
SPF-MPF	0.1574	100.00%	0.0052	97.12%	0.0071	96.74%
Marginal BPF	0.9876	0.21%	0.2641	1.79%	0.1723	12.60%
Marginal EPF	0.7857	1.90%	0.3097	16.69%	0.1820	28.63%
Marginal UPF	0.7870	2.04%	0.3112	14.46%	0.1674	25.32%
Marginal APF	0.0670	100.00%	0.0153	92.92%	0.0596	72.00%

Table 2.4: Comparison of densities for the univariate examples

Linear, Bimodal, Bivariate Model

This example poses a bimodal model where the modes arise from two different observations with a joint likelihood explicitly known. In the algorithm for propagating particles, we implemented a scheme that preselects samples to be filtered for either observation. This is done according to a set of indexes that are sampled from a binomial distribution $B\left(u_1, u_2; 1, w_{l,(1)}, w_{l,(2)}\right) \propto w_{l,(1)}^{u_1} w_{l,(2)}^{u_2}$ where $u_1, u_2 \in [0,1]$, $u_1 + u_2 = 1$, such that indexes are uniquely associated to either event u_1 or u_2, with probability of either observation, $w_{l,(1)}$ or $w_{l,(2)}$ respectively. The linear, bimodal, bivariate model was tested with parameters set as shown in Table 2.5. These parameters were chosen to result in quite distinct local properties of the two modes.

Parameters for the linear, bimodal, bivariate model	
Initial distribution	$\bar{\mathrm{x}}_{k-1} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$, $\mathrm{P}_{k-1} = \begin{pmatrix} 9 & 0 \\ 0 & 9 \end{pmatrix}$
Markov transition pdf	$f(\mathrm{x}_{k-1}) = \mathrm{x}_{k-1}$, $\mathrm{Q}_k = \begin{pmatrix} 16 & 0 \\ 0 & 16 \end{pmatrix}$
Likelihood function:	$h(\mathrm{x}_k) = \mathrm{x}_k$
Mode 1	$\mathrm{R}_{k,(1)} = \begin{pmatrix} 0.8 & 0 \\ 0 & 0.2 \end{pmatrix}$, $w_{l,(1)} = 0.2$
Mode 2	$\mathrm{R}_{k,(2)} = \begin{pmatrix} 4.0 & 0 \\ 0 & 1.0 \end{pmatrix}$, $w_{l,(2)} = 0.8$
Observations	$\mathrm{y}_{k,(1)} = \begin{pmatrix} +10 \\ +20 \end{pmatrix}$, $\mathrm{y}_{k,(2)} = \begin{pmatrix} +10 \\ -20 \end{pmatrix}$

Table 2.5: Parameters for the bimodal bivariate model

For this example, we analyze stochastic particle flow methods, SPF-GS and SPF-MPF, against original particle flow methods only. We exemplify the sequence of particles' distributions acquired by the GPF, by the SDPF and by the stochastic particle flow in Figure 2.7. It becomes clear that the final distribution generated by the stochastic particle flow is closely similar to the true posterior density, precisely describing the local moments of the two modes. In opposition, the GPF generates a distribution that is excessively biased for the most peaky mode whereas the SDPF generates a distribution that does not describe correctly the covariances of each mode.

These findings are quantified by the average Jensen-Shannon divergences presented in Table 2.8. Table 2.8 shows a small divergence of the density filtered by the SPF-GS with respect to the true posterior, a small divergence of the SPF-MPF proposal density as well as of the target

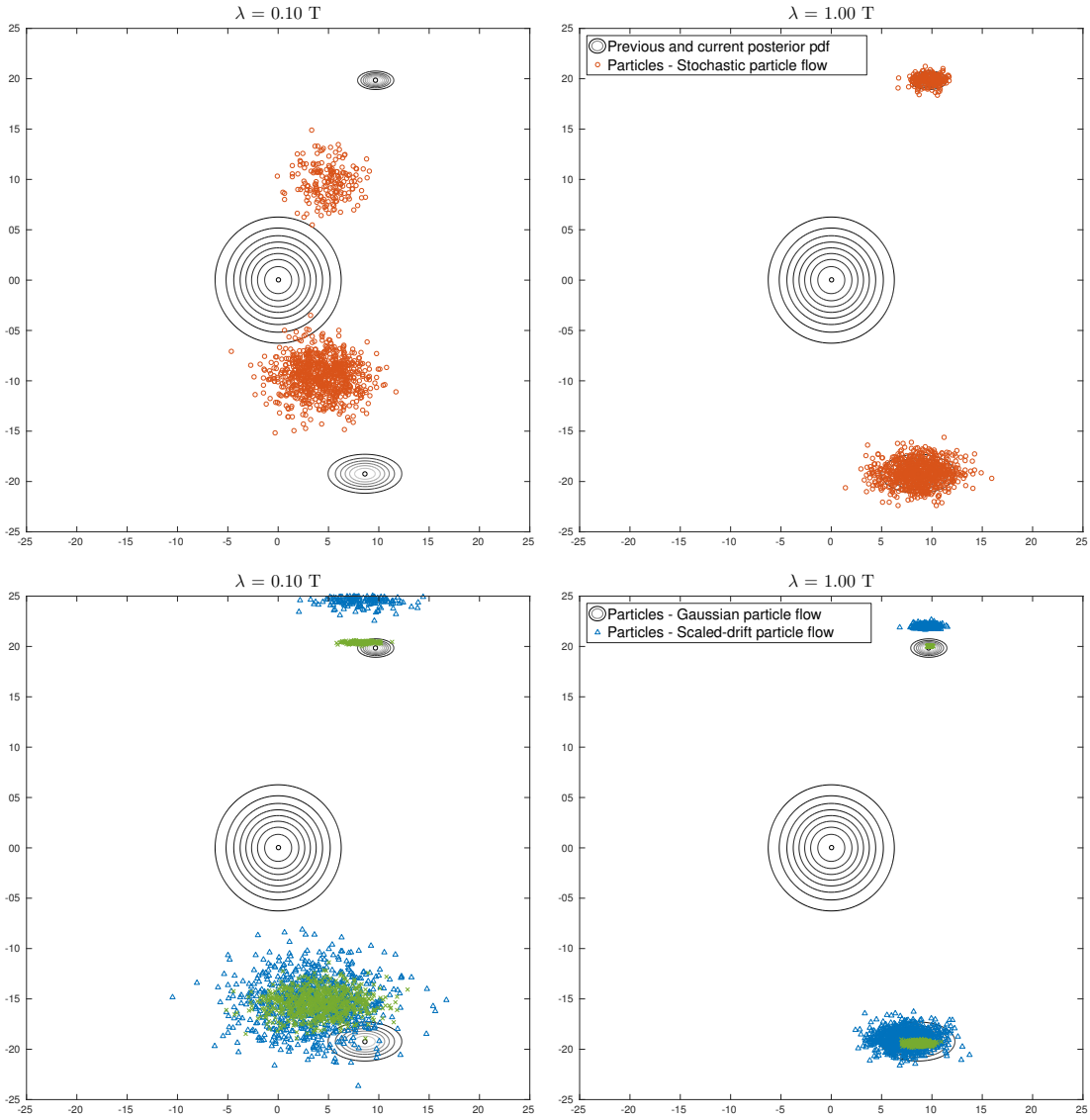


Figure 2.7: Sequence of samples' distributions for the bimodal, bivariate example

density, whereas the divergences of the original particle flows are fairly big. The SPF-MPF provides a high effective sample size.

Nonlinear, Unimodal, Bivariate Model

The nonlinear bivariate model was tested in two cases:

Case 1. favorable for marginal particle filters, and

Case 2. unfavorable, i.e., emulating a scenario similar to that presented in the first toy example where importance sampling fails.

The parameters used for Cases *1* and *2* are presented in Table 2.6 and Table 2.7, respectively.

In either cases the sequence of distributions generated by the original particle flows and by the stochastic particle flow are as illustrated in Figure 2.8. Once more it becomes evident that the stochastic particle flow provides a superior distribution of samples to approximate

Parameters for the nonlinear bivariate model, Case 1	
Initial distribution	$\bar{\mathrm{x}}_{k-1} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$, $\mathrm{P}_{k-1} = \begin{pmatrix} 20 & 0 \\ 0 & 20 \end{pmatrix}$
Markov transition pdf	$f(\mathrm{x}_{k-1}) = \mathrm{x}_{k-1}$, $\mathrm{Q}_k = \begin{pmatrix} 20 & 0 \\ 0 & 20 \end{pmatrix}$
Likelihood function	$h(\mathrm{x}_k) = \begin{pmatrix} \sqrt{\mathrm{x}_k(1)^2 + \mathrm{x}_k(2)^2} \\ \operatorname{atan}\left(\mathrm{x}_k(2)/\mathrm{x}_k(1)\right) \end{pmatrix}$, $\mathrm{R}_k = \begin{pmatrix} 1.00 & 0 \\ 0 & 0.16 \end{pmatrix}$
Observation	$\mathrm{y}_k = \begin{pmatrix} 20 \\ 0^\circ \end{pmatrix}$

Table 2.6: Parameters for the nonlinear bivariate model, Case 1

the posterior density, which demonstrates its higher level of accuracy and regularity. Similarly to results presented for previous examples, the GPF seems to generate substantially biased distributions whereas the SDPF seems highly prone to regularity problems. These aspects are well corroborated by the average Jensen-Shannon divergences presented in Table 2.8.

In the comparison we also included other marginal particle filters. For *Case 1* (favorable), we illustrate in Figures 2.9 and 2.10 how the marginal importance densities, projected (marginalized) onto the horizontal and vertical planes, would look like as proposed by the marginal auxiliary particle filter (MAPF) and by the SPF-MPF. It is clear that in this case both MAPF and SPF-MPF generate proposal densities quite proximate of the empirical marginal target, which in turn approximates well the true posterior. Additionally, it is possible to visualize that the SPF-MPF provides a slightly better proposal density in terms of similarity to the target density, which is corroborated by a greater average effective sample size as presented in Table 2.8. All other marginal particle filters don't generate effective importance densities in terms of approximating either the true posterior or the target density.

For *Case 2* (unfavorable), importance sampling fails as exemplified by the projections of the importance density proposed by the MAPF depicted in Figure 2.11. By the same reason explained before, the importance sampling procedure fails to provide a satisfactory filtering measure owing to the errors that affect evaluations of both the marginal target density and the marginal importance density. As a consequence, in this case, any marginal particle filter generates a poor solution, although the MAPF provides a high effective sample size. The SPF-MPF generates a remarkably poor solution for *Case 2* because it distributes particles to approximate the true posterior density by design but must constrain the proposal mixture components to match a very inaccurate empirical target density.

In contrast, in both Cases *1* and *2* the SPF-GS proposes a direct filtering density that accurately approximates the true posterior density. The SPF-GS is demonstrated to be insensitive to the issues caused by an observation located relatively far from the initial distribution. These features are quantitatively captured by the performance indexes summarized in Table 2.8.

Parameters for the nonlinear bivariate model, Case 2	
Initial distribution	$\bar{\mathrm{x}}_{k-1} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$, $\mathrm{P}_{k-1} = \begin{pmatrix} 10 & 0 \\ 0 & 10 \end{pmatrix}$
Markov transition pdf	$f(\mathrm{x}_{k-1}) = \mathrm{x}_{k-1}$, $\mathrm{Q}_k = \begin{pmatrix} 5 & 0 \\ 0 & 5 \end{pmatrix}$
Likelihood function	$h(\mathrm{x}_k) = \begin{pmatrix} \sqrt{\mathrm{x}_k(1)^2 + \mathrm{x}_k(2)^2} \\ \mathrm{atan}\left(\mathrm{x}_k(2)/\mathrm{x}_k(1)\right) \end{pmatrix}$, $\mathrm{R}_k = \begin{pmatrix} 1.00 & 0 \\ 0 & 0.16 \end{pmatrix}$
Observation	$\mathrm{y}_k = \begin{pmatrix} 20 \\ 0^\circ \end{pmatrix}$

Table 2.7: Parameters for the nonlinear bivariate model, Case 2

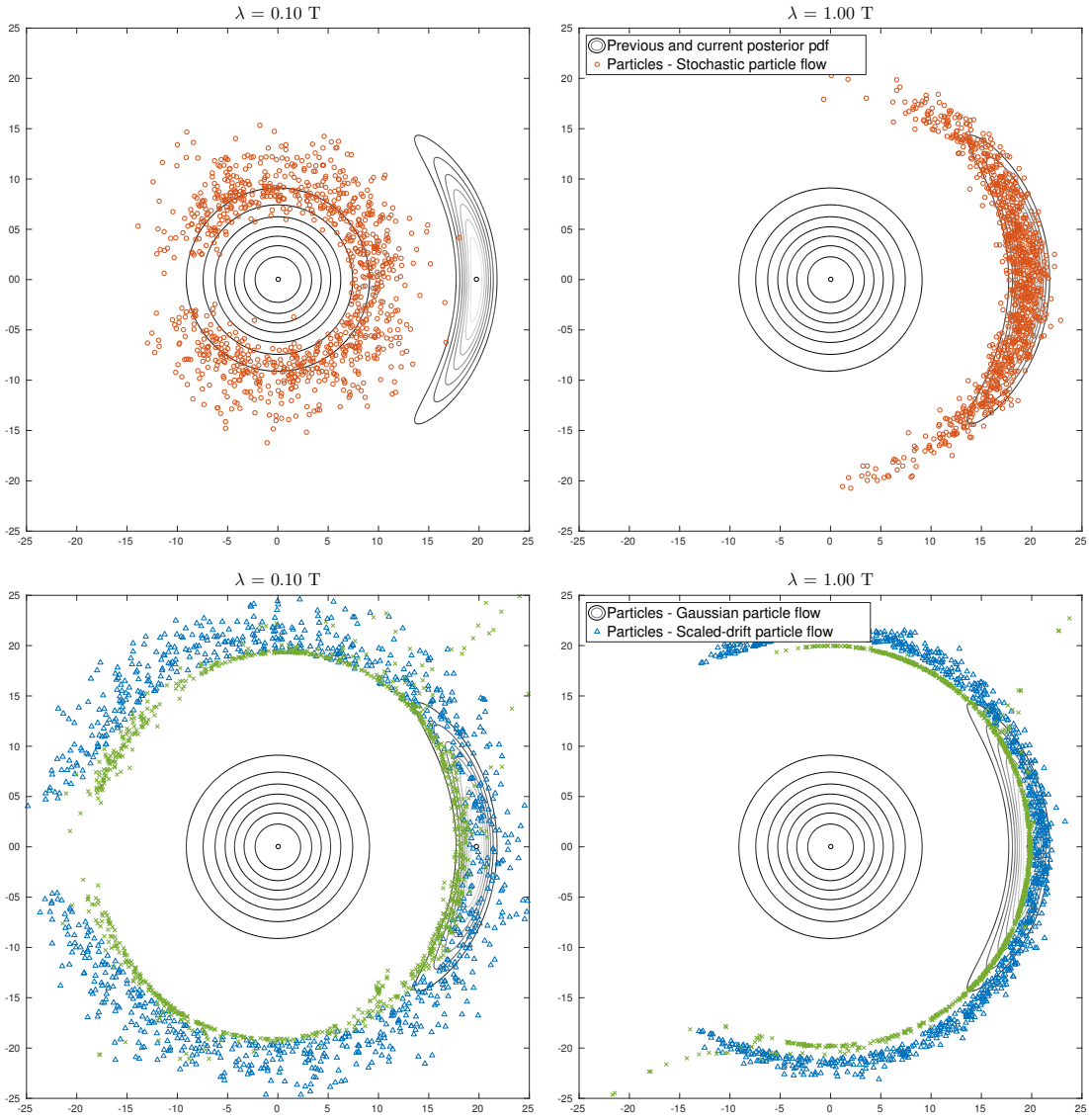
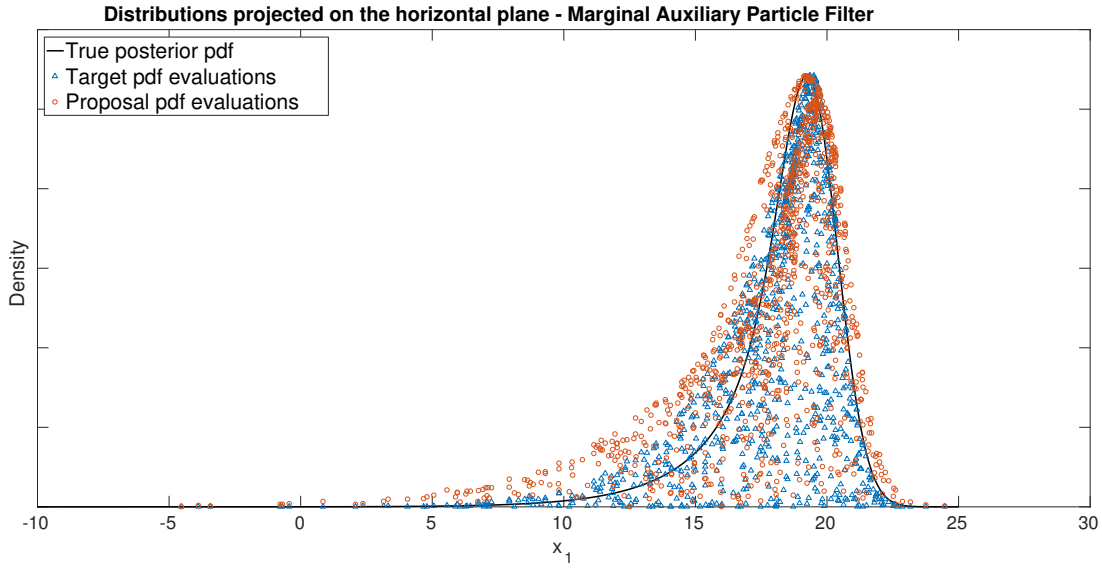
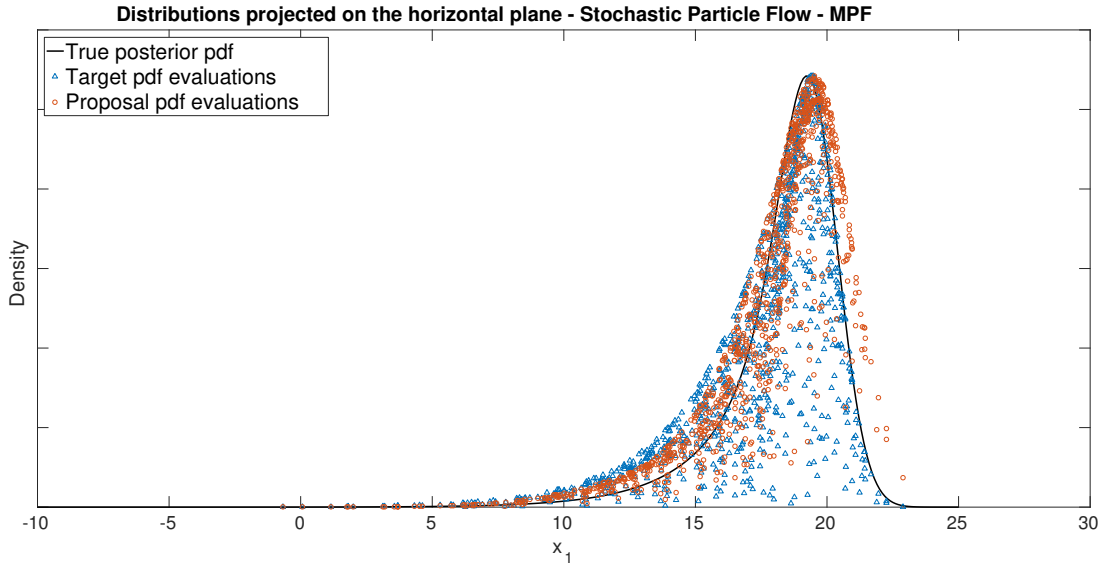


Figure 2.8: Sequence of samples' distributions for the nonlinear, bivariate example



(a) Marginal Auxiliary Particle Filter

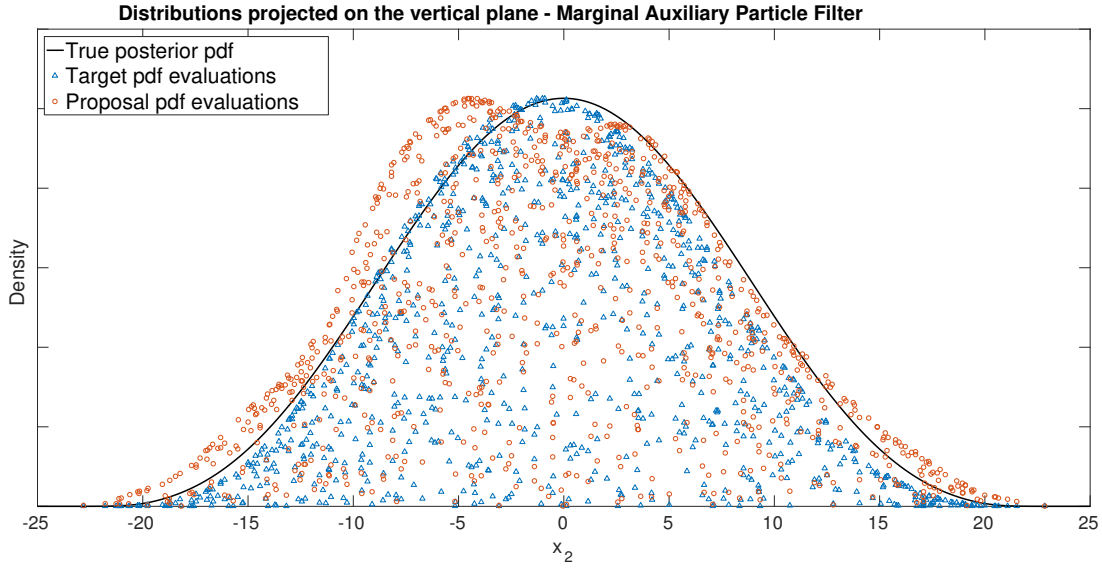


(b) Stochastic Particle Flow - MPF

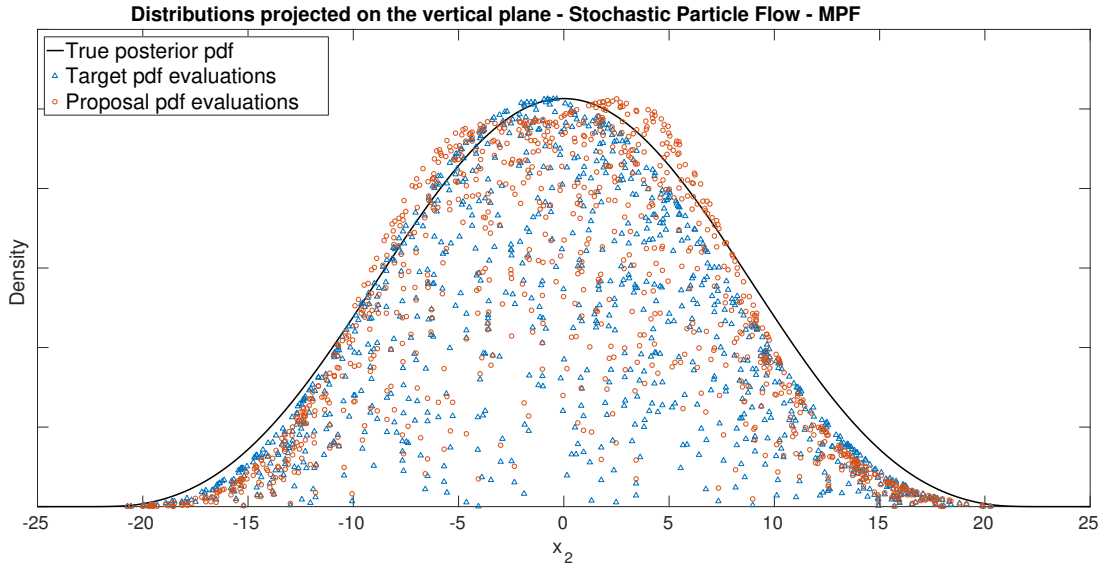
Figure 2.9: Horizontal-plane projection of densities for the nonlinear, bivariate example

Density	*Multimodal, linear*		*Nonlinear - Case 1*		*Nonlinear - Case 2*	
	JSD_{avg}	ESS_{avg}	JSD_{avg}	ESS_{avg}	JSD_{avg}	ESS_{avg}
Marginal target	0.0118	-	0.0074	-	0.2444	-
SPF-GS	0.0003	-	0.0133	-	0.0755	-
SPF-MPF	0.0217	93.00%	0.0112	84.01%	0.2746	10.79%
Gaussian particle flow	0.2647	-	0.6563	-	0.5279	-
Scaled-drift particle flow	0.3866	-	0.4962	-	0.5804	-
Marginal BPF	-	-	0.9969	0.37%	0.9998	0.13%
Marginal EPF	-	-	0.3131	27.57%	0.5714	8.83%
Marginal UPF	-	-	0.7753	4.44%	0.8136	2.31%
Marginal APF	-	-	0.0119	81.32%	0.1467	85.11%

Table 2.8: Comparison of densities for the bivariate examples

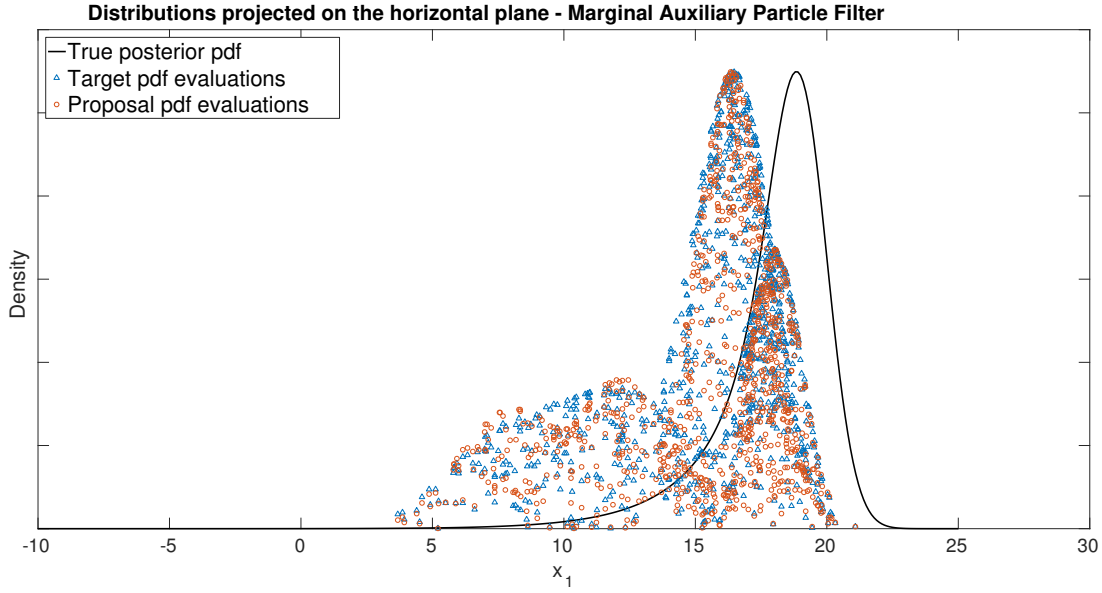


(a) Marginal Auxiliary Particle Filter

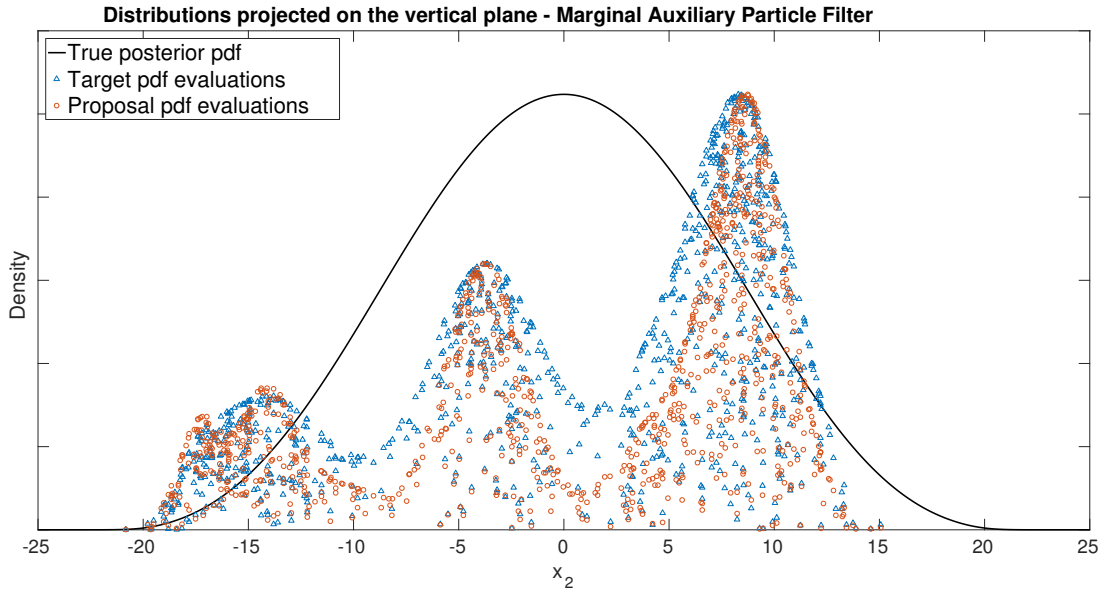


(b) Stochastic Particle Flow - MPF

Figure 2.10: Vertical-plane projection of densities for the nonlinear, bivariate example



(a) Horizontal plane



(b) Vertical plane

Figure 2.11: Failure of marginal importance sampling for the nonlinear, bivariate example

2.7.2 Multi-Sensor Bearings-Only Tracking

Estimation in clutter of a target's position and velocity based solely on angular measurements is a relevant problem that finds direct application in airborne radar and sonar in passive listening mode. We propose an example where a single target is observed by a circumferential array of sensors. Each sensor measures the target's bearing with respect to its own position.

In this example, we compare performances of the following filters:

- multi-sensor EKF that performs a series of centralized measurement updates, considering each sensor in sequence;
- Information Matrix Fusion filter (IMF-EKF) [7] that fuses distributed estimates (in parallel) into a global estimate using the Information Matrix form of the EKF;
- bootstrap particle filter (SIR);
- Gaussian particle flow (GPF);
- scaled-drift particle flow (SDPF); and
- stochastic-particle-flow Gaussian sum (SPF-GS).

As mentioned before, in order to work properly, both the GPF and SDPF are implemented based on a companion filter that estimates the state covariance matrix. This is in accordance with implementations suggested by Choi *et al.* [30] and Ding & Coates [63]. In this example we used the multi-sensor EKF as companion filter for both the GPF and SDPF. In contrast, the stochastic particle flow does not require a companion filter.

In all multi-sensor and multi-target tracking examples we shall use the root-mean-square error (RMSE) and the normalized-estimation error squared (NEES) as performance indexes, hence, for convenience, we provide their definition as follows. Suppose that the sequence of true d_x-dimensional states (ground truth) of a moving object is known, $\{\mathrm{x}_{k,\mathrm{truth}}\}_{k\in[0..K]}$. Provided that measurements are generated, $\{\mathrm{y}_k^{(r)}\}_{k\in[1..K],r\in[1..N_{\mathrm{MC}}]}$, and estimates (along with state covariances) are produced by a filter, $\{\hat{\mathrm{x}}_k^{(r)},\hat{\mathrm{P}}_k^{(r)}\}_{k\in[1..K],r\in[1..N_{\mathrm{MC}}]}$, over N_{MC} Monte Carlo runs, the RMSE and the NEES are defined respectively as

$$\mathrm{RMSE}_k \triangleq \sqrt{\frac{1}{N_{\mathrm{MC}}}\sum_{r=1}^{N_{\mathrm{MC}}}(\hat{\mathrm{x}}_k^{(r)}-\mathrm{x}_{k,\mathrm{truth}})^{\mathrm{T}}(\hat{\mathrm{x}}_k^{(r)}-\mathrm{x}_{k,\mathrm{truth}})}, \tag{2.81}$$

$$\mathrm{NEES}_k \triangleq \frac{1}{d_x N_{\mathrm{MC}}}\sum_{r=1}^{N_{\mathrm{MC}}}(\hat{\mathrm{x}}_k^{(r)}-\mathrm{x}_{k,\mathrm{truth}})^{\mathrm{T}}\hat{\mathrm{P}}_k^{(r)\,-1}(\hat{\mathrm{x}}_k^{(r)}-\mathrm{x}_{k,\mathrm{truth}}), \tag{2.82}$$

The RMSE indicates how accurate the estimates provided by a filter are, in the mean square sense, and the NEES indicates how consistently (or credibly) a filter can produce estimates, in the sense that the estimates variability as given by consistent estimators should be small over many trials. Ideal estimators should produce accurate and consistent estimates, that is, $\mathrm{RMSE}_k \approx 0$ and $\mathrm{NEES}_k \approx 1$ from above[10] for $k = 1,\ldots,K$.

[10] It is expected that $\frac{1}{N_{\mathrm{MC}}}\sum_{r=1}^{N_{\mathrm{MC}}}(\hat{\mathrm{x}}_k^{(r)}-\mathrm{x}_{k,\mathrm{truth}})^{\mathrm{T}}\hat{\mathrm{P}}_k^{(r)\,-1}(\hat{\mathrm{x}}_k^{(r)}-\mathrm{x}_{k,\mathrm{truth}}) \searrow \mathrm{tr}\{\mathbb{I}_{d_x}\} = d_x$ for estimators increasingly consistent.

The bootstrap particle filter (SIR), the GPF and SDPF, and the SPF-GS consider all measurements jointly according to a joint likelihood function described in the next section. Performance is analyzed by computing the root-mean-square error (RMSE) of estimates and the normalized-estimation error squared (NEES) over 100 Monte Carlo runs. All particle-based filters use 200 samples.

A Multi-Sensor Bearings-Only Model

When tracking in clutter based on multiple measurements, the usual treatment rests on the probabilistic data association (PDA) [6]. In the PDA model, a set of m_k valid measurements is received at each time step k and assumed to be generated according to the possibilities: (i) all measurements are false alarms (clutter), (ii) one of the measurements is originated from the target and the remaining are false alarms. Let $\theta_{k,i}$ be the association event that the ith measurement is target-originated. The PDA filter computes the association probabilities $p(\theta_{k,i}|\mathrm{y}_{1:k})$ conditional on the set of all received measurements up to time instant k, and calculates the target state posterior density, $p(\mathrm{x}_k|\mathrm{y}_{1:k})$, by marginalizing the joint density $p(\mathrm{x}_k, \theta_{k,1:m_k}|\mathrm{y}_{1:k})$ over all possible associations.

In our example, a single target is tracked by a set of N_s sensors located along a circumference that encloses the surveillance region, at equally-spaced angular positions. As per the PDA model, one target is known to exist a priori, detected with probability $P_{d,j}$ by the jth sensor, and the number of clutter detections per sensor is Poisson-distributed with mean $\lambda_c \cdot V$, where λ_c is the clutter spatial density and V is the surveillance region's volume. For any given set of N_s sensors, the expected likelihood can be easily obtained by extending the procedure established by Marrs *et al.* [133] to multiple sensors, to give

$$p\left(\mathrm{y}_k|\mathrm{x}_k, \mathrm{y}_{1:k-1}\right) = \prod_{j=1}^{N_s} V^{-m_{k,j}} \frac{(\lambda_c V)^{m_{k,j}}\, e^{-\lambda_c V}}{m_{k,j}!} \left[\lambda_c(1-P_{d,j}) + \sum_{i=1}^{m_{k,j}} P_{d,j}\mathcal{N}(\mathrm{y}_{k,i(j)}; h_j(\mathrm{x}_k), \mathrm{R}_{k,j})\right], \quad (2.83)$$

where $m_{k,j}$ is the total number of validated measurements for the jth sensor, $\mathrm{y}_{k,i(j)}$ is the ith measurement received by the jth sensor, $h_j(\cdot)$ and $\mathrm{R}_{k,j}$ are the observation function and the observation noise variance for the jth sensor, respectively.

On a bidimensional state space, the bearing observations are modeled by

$$h_j(\mathrm{x}_k) = \mathrm{atan}\left(\frac{\mathrm{x}_k\,(2) - p_{2,j}}{\mathrm{x}_k\,(1) - p_{1,j}}\right), \quad (2.84)$$

where $\mathrm{p}_j = (p_{1,j}, p_{2,j})^T$ are the position coordinates for the jth sensor. We assume a target moving according to the nearly-constant velocity model,

$$\mathrm{x}_k = \mathrm{F}\mathrm{x}_{k-1} + \mathrm{u}_k,\ \mathrm{u}_k \sim \mathcal{N}(\mathrm{u}_k; 0_{d_x}, \mathrm{Q}_k), \quad (2.85)$$

where $\mathrm{x}_k = (p_{x_1}, p_{x_2}, v_{x_1}, v_{x_2})_k^T$ is the state vector composed of position and velocity in Cartesian

coordinates (x_1, x_2), and

$$\mathrm{F} = \begin{pmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \tag{2.86}$$

$$\mathrm{Q}_k = \begin{pmatrix} {\Delta t^3}/{3} & 0 & {\Delta t^2}/{2} & 0 \\ 0 & {\Delta t^3}/{3} & 0 & {\Delta t^2}/{2} \\ {\Delta t^2}/{2} & 0 & \Delta t & 0 \\ 0 & {\Delta t^2}/{2} & 0 & \Delta t \end{pmatrix} \sigma_q^2. \tag{2.87}$$

The multi-sensor joint likelihood (2.83) is incorporated in the bootstrap (SIR) filter, the GPF and SDPF, and the SPF-GS by considering their filtering densities to target a posterior density involving such joint likelihood. Regarding the implementation of particle flows, specifically for this problem, the GPF and SDPF reinterpret the filtered density empirically as a Gaussian pdf at the end of each iteration in order to avoid exponential growth of the number of mixture components over time. This practical aspect does not affect the SPF-GS, whose filtered density is a mixture composed invariably of N local solutions to the actual posterior pdf, where N is the number of samples (and mixture components).

Generally speaking, multi-sensor bearings-only tracking is a difficult problem to solve when the observation noise has high variance, the probability of detection is relatively low and the probability of having clutter in the surveillance region is not negligible. In this scenario, the difficulty stems from the fact that the joint multi-sensor likelihood (2.83) is a product of mixtures composed of several nonlinear and non-informative likelihood terms: when nonlinearity is pronounced by a high-variance observation noise, the resulting posterior density may not be well expressed by simple parametric densities. In addition, this difficulty is modulated by the amount of information available: the fewer the number of sensors the more difficult to solve the problem. Another aspect that poses additional concern is the system's observability. It is highly dependent on the relative position of a sensor with respect to the target's trajectory, i.e., trajectories radially aligned with a sensor's position provide less information on the target's velocity.

Results

A challenging scenario was set for comparing the filters in order to exacerbate differences in their performances to a noticeable level. In this very difficult scenario, state process noise is assumed with variance scaled by $\sigma_q^2 = 25\,(\mathrm{m/s})^2/\mathrm{s}$, observation noise variance is $\mathrm{R}_k = 25\,\mathrm{deg}^2$, $P_d = 0.50$ and $\lambda_c \times V = 1.00$ false alarm/sensor/scan, for identical sensors. Even though estimation errors generated for such a scenario might not be feasible as an Engineering solution, it is certainly of practical interest to examine how the estimates' errors scale to such extreme scenarios, which might happen in real applications. This problem is particularly interesting because the smaller the number of sensors the more difficult to achieve reasonable estimates since, in this case, the low signal-to-noise ratio would deteriorate the inference. For this example, the SPF-GS has been set with time horizon $T = 10\,\mathrm{s}$ and integration step size $\Delta\lambda = 1\,\mathrm{s}$.

The resulting track of an exemplary run is shown in Figure 2.12. The track initiation is

based on an overdetermined triangulation of measurements for the first two steps. No gating has been performed, i.e., no preprocessing to discard measurements that fall outside a high-confidence region of each sensor. Figure 2.12 depicts a successful tracking of the target despite the difficult scenario.

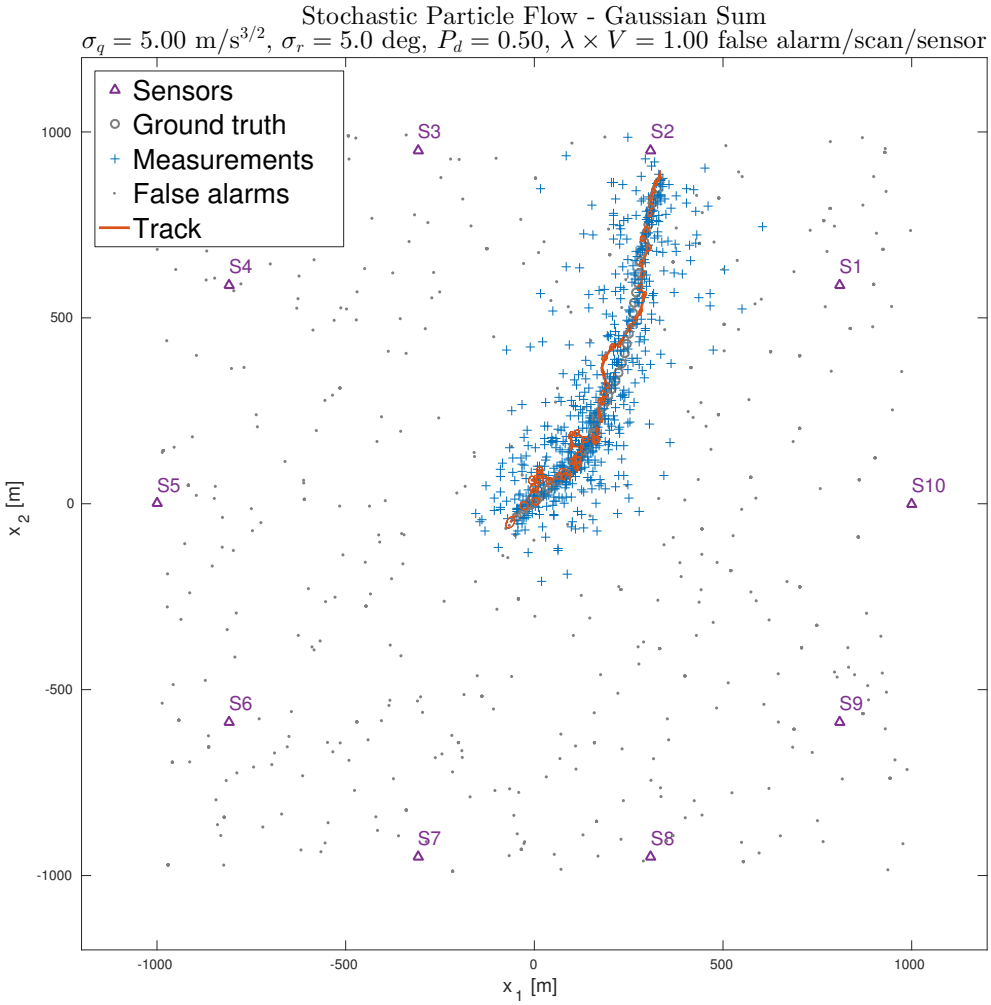


Figure 2.12: Illustration of bearing-only multi-sensor tracking

The resulting root-mean-square error (RMSE), normalized-estimation error squared (NEES), and average computation time (per time step) of all filters for different numbers of sensors are shown in Figures 2.13, 2.14 and 2.15 respectively. The following important aspects can be observed from Figure 2.13:

- Somewhat counterintuitively, the multi-sensor serial EKF provides better estimates than that of Information Matrix Fusion EKF (IMF-EKF), both in terms of precision (RMSE) and "consistency" (or credibility[11] as indicated by NEES).
- The scaled-drift particle flow (SDPF) shows remarkably poor performance.
- The bootstrap particle filter (SIR) provides mediocre performance, eventually becoming better than the IMF-EKF as the number of sensors increases.

[11] As presented in [122], the normalized-estimation error squared (NEES) is the simplest metric that indicates an estimator's credibility.

- Accuracies shown by the Gaussian particle flow (GPF) and the serial EKF are commensurate and similar to that of the stochastic particle flow (SPF-GS) when the number of sensors is high.
- As expected, the overall estimation accuracy is improved as the number of sensors is increased.
- The SPF-GS provides the most accurate estimates in difficult scenarios, i.e., when the number of sensors is small.
- Estimation by the SPF-GS is more consistent (or credible), which is denoted by a NEES closer to one ($\log_{10}\text{NEES} \to 0$) from above.

It is worth commenting on the results comparing GPF and SPF-GS. Specifically for this problem, when the number of sensors is sufficiently high, the GPF provides estimates as accurate as those of SPF-GS at a slightly lower computational cost. It is also remarkable the successful synergy between the GPF and its companion filter, a multi-sensor serial EKF that provides the covariance estimates. However, it is difficult to justify the calculated NEES for original particle flows since their first and second moment estimates are underpinned by distinct filtering methods. On the other hand, the stochastic particle flow (SPF-GS) provides fairly accurate state estimates and securely constitutes the most credible estimator among all evaluated filters.

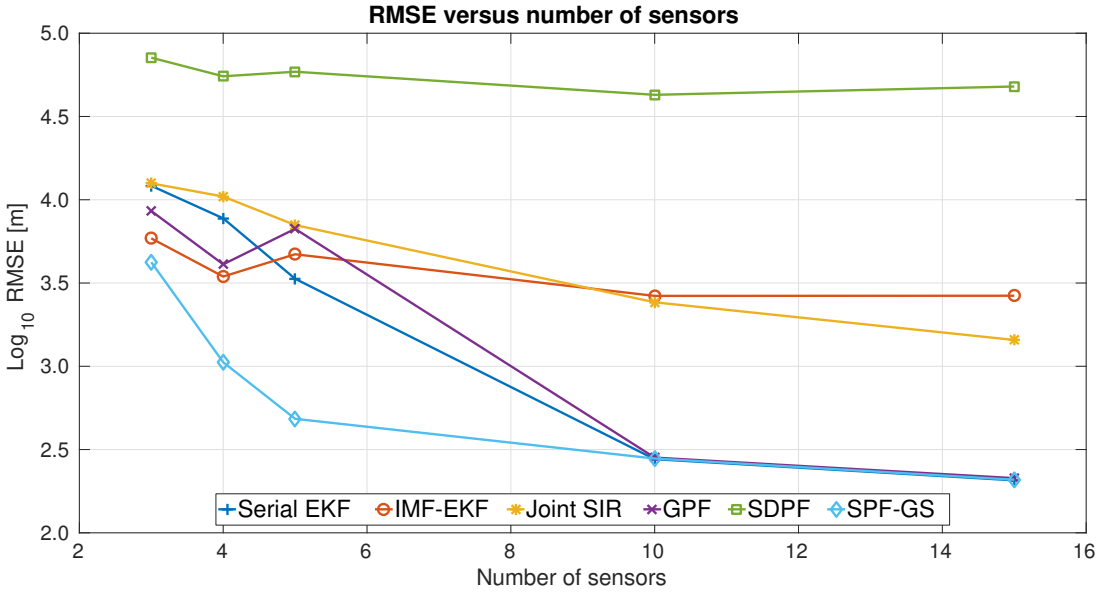


Figure 2.13: RMSE for the multi-sensor bearing-only tracking example

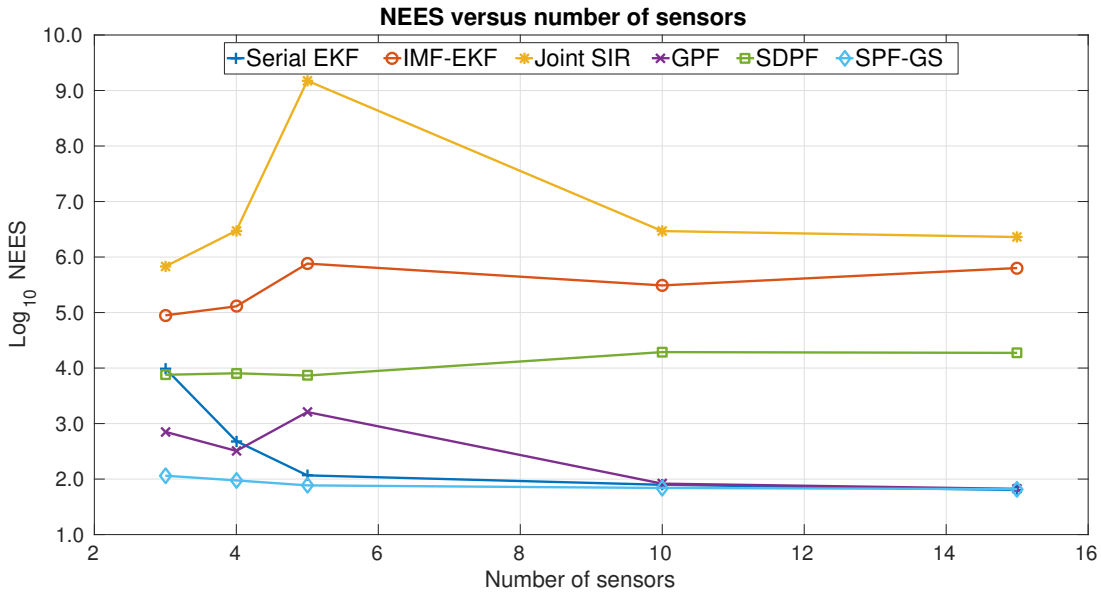


Figure 2.14: NEES for the multi-sensor bearing-only tracking example

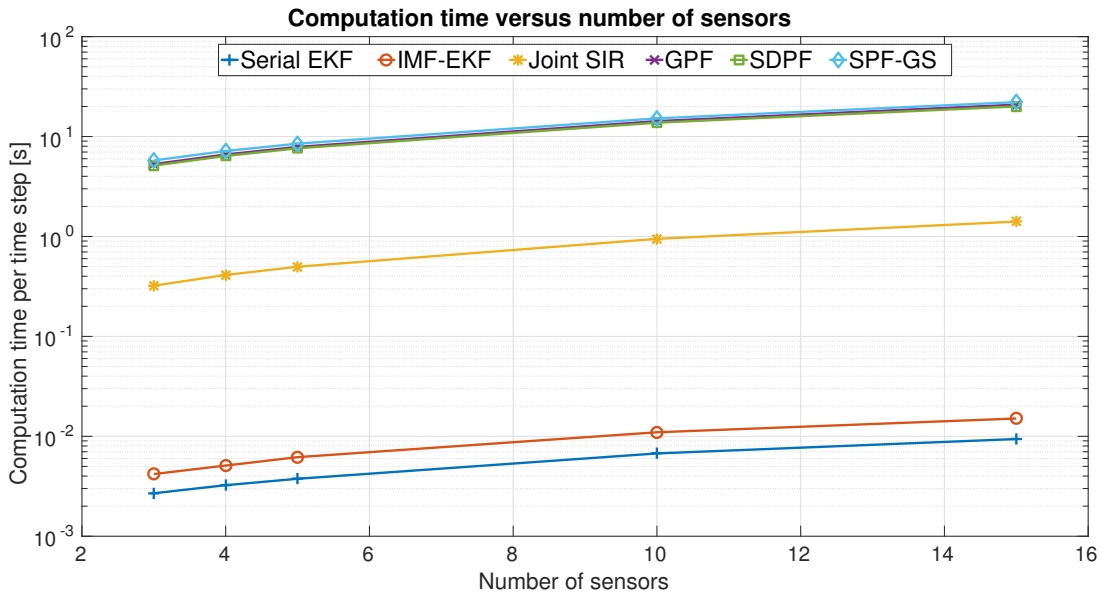


Figure 2.15: Average computation time for the multi-sensor bearing-only tracking example

2.7.3 Convoy Tracking

Tracking multiple objects in clutter is as challenging as important for real applications. In the multi-target tracking standard methods, the most common treatment assumes the targets' states to be independent so that the joint probability density is the product of their marginal densities. While this assumption is fairly reasonable for applications where objects are far apart most of the time, the same cannot be stated for cases where objects are in proximity for a considerable part of time. These latter cases elicit tracking all targets jointly in the hope of implicitly capturing dependencies between targets. When targets are tracked jointly the problem's dimensionality scales with the number of targets.

To illustrate this situation, we propose an example of a convoy of vehicles that are forced into mutual proximity when trafficking on a road. The vehicles have explicit interactions as each driver aims driving at the maximum allowed speed unless there is another vehicle immediately in front at a slower speed. This scenario requires care for a safe distance. The intent of the tracker is then to provide the best estimate of each vehicle on a convoy, given a set of non-identified measurements corrupted by noise and possible false alarms (clutter) reported by a position sensor.

We compare performances of the following filters:

- Joint Probabilistic Data Association (JPDA) filter [6];
- Global Nearest Neighbor Data Association (GNN) filter;
- JPDA with a Gaussian mixture per target (JPDA-GM), applying mixture reduction [153];
- Coupled Probabilistic Data Association (CPDA) filter [16];
- multi-target bootstrap particle filter (joint SIR) based on the description by Blom & Bloem [15];
- Gaussian particle flow (GPF);
- scaled-drift particle flow (SDPF); and
- stochastic-particle-flow Gaussian sum (SPF-GS).

The GPF and the SDPF rely on a companion filter to estimate state covariance matrices correctly, according to implementation guidelines by Choi *et al.* [30] and Ding & Coates [63]. In this example, we used the CPDA [16] as companion filter for the original particle flows (GPF, SDPF). In contrast, the SPF-GS does not require a companion filter.

The CPDA, the joint bootstrap particle filter (SIR), the GPF and SDPF, and the SPF-GS consider all targets' states jointly, as a single high-dimensional state. In contrast, the classical multi-target filters track targets separately, where each target's state is described by the nearly-constant velocity model. Performance is analyzed by computing the root-mean-square error (RMSE) of estimates and the normalized-estimation error squared (NEES) over 100 Monte Carlo runs. The particle filter (SIR), the original particle flows and the stochastic particle flow use 200 samples.

The Intelligent Driver Model

The Intelligent Driver Model (IDM) [193] is a model[12] used in Traffic Engineering to simulate phenomena such as congestion and to analyze the traffic behavior as a response to changes in the transport system. Because the interaction between vehicles is explicitly taken into account by the IDM, tracking based on it involves consideration of the joint state of multiple targets. Even though the IDM establishes an empirical description of traffic for multiple vehicles, it has not been previously used in the context of multi-target tracking. We propose a stochastic

[12] In its simplest form, the IDM is focused on the interaction of vehicles moving along a single-carriageway road. More complex variants exist to model overtaking, for example, and consider factors such as the politeness of the driver.

version of the IDM and discretize it in order to make it compatible with multi-target trackers formulated on the joint state space.

The IDM describes the dynamics of vehicles in traffic, in terms of positions and velocities, incorporating the interaction between each vehicle and the vehicle directly in front. Provided a vehicle indexed as α with length l_α, the dynamics of its position x_α and velocity v_α are given by the following (continuous-time) stochastic differential equations:

$$dx_\alpha = v_\alpha dt, \tag{2.88}$$

$$dv_\alpha = \underbrace{a\left[1 - \left(\frac{v_\alpha}{v_0}\right)^\delta\right]}_{\dot{v}_\alpha^{\text{free road}}} dt - \underbrace{a\left[\frac{\bar{s}}{s_\alpha}\right]^2}_{\dot{v}_\alpha^{\text{interaction}}} dt + dw_t, \tag{2.89}$$

where $\{w_t\}_{t\geq 0}$ is a Wiener process, $s_\alpha = x_{\alpha-1} - x_\alpha - l_\alpha$ is the net distance between vehicles, the approaching rate is given by $\Delta v_\alpha = v_{\alpha-1} - v_\alpha$, and $\bar{s} = \bar{s}\,(v_\alpha, \Delta v_\alpha)$ is the expected distance defined as

$$\bar{s}\,(v_\alpha, \Delta v_\alpha) = s_0 + v_\alpha T_h + \frac{v_\alpha \Delta v_\alpha}{2\sqrt{a \cdot b}}. \tag{2.90}$$

The model dynamics is such that when a vehicle is travelling on a free road it will predominantly accelerate according to $\dot{v}_\alpha^{\text{free road}}$ up to the maximum allowed speed v_0, whereas when it approaches another vehicle immediately in front, the decrement in acceleration according to $\dot{v}_\alpha^{\text{interaction}}$ becomes relevant to maintain a safe-time headway T_h and to avoid approaching closer than the minimum safe distance s_0. The IDM parameters are summarized in the following table.

Parameter	**Description**
a	nominal maximum acceleration
b	comfortable braking deceleration
δ	acceleration exponent (driver dynamics)
v_0	free-road desired velocity
s_0	minimum allowed distance between vehicles
T_h	safe-time headway
$\alpha - 1$	index of the vehicle direcly in front

Table 2.9: Parameters of the Intelligent Driver Model (IDM)

In order to use the stochastic IDM as the state process for a multi-target tracker, its continuous-time equations are discretized by a first-order approximation (Markov random field). This assumes that the state's derivative with respect to time is linear in time between two subsequent measurements but the interactions between non-adjacent vehicles are negligible when compared to the interactions between adjacent vehicles. The discretized version of the stochastic IDM is presented in Section 2.9.5.

The Multi-Target Likelihood Function

The joint multi-target filters extend the joint probabilistic data association (JPDA) [6] framework for situations where the targets' states are not mutually independent conditioned on the past observations. This formulation has been first proposed as the JPDA Coupled filter

(JPDAC) [6] and further generalized by Blom & Bloem [16, 15] who consider the measurement-to-target associations implicitly.

In the JPDA model, a set of N_m valid measurements is received at each time step k and assumed to be generated according to the possibilities: (i) each of the measurements may be originated from each target, considering all possible associations, (ii) a measurement not originated from any target is due to a false alarm (clutter). These possibilities are exhaustive such that a measurement can have only one source, and at most one of the validated measurements can originate from a target.

Let $\phi_{k,i} \in \{0, 1, \ldots, N_m\}$ be an association event that maps each target $i \in \{1, \ldots, N_t\}$ to the measurement indexed as $\phi_{k,i}$, where $\phi_{k,i} = 0$ means that no measurement is associated to the ith target. The Coupled JPDA filter computes the joint association probabilities $p(\phi_{k,1:N_t}|\mathrm{y}_{1:k})$ conditional on the set of all received measurements up to time instant k, and calculates the joint state posterior density, $p(\mathrm{x}_{k,1:N_t}|\mathrm{y}_{1:k})$, by marginalizing $p(\mathrm{x}_{k,1:N_t}, \phi_{k,1:N_t}|\mathrm{y}_{1:k})$ over all possible joint associations.

In the JPDA framework, N_t targets are known to exist a priori, detected with probability P_d by a single sensor; the number of clutter detections is Poisson-distributed with mean $\lambda_c \cdot V$, where λ_c is the clutter spatial density and V is the surveillance region's volume; the location of each clutter detection is independently distributed according to a spatial density $\eta_c(\mathrm{y})$; and the likelihood function of the jth measurement being originated from the ith detected target is $p(\mathrm{y}_{k,j}|\mathrm{x}_{k,i})$. Denoting the joint multi-target state as $\mathrm{x}_{k,1:N_t}$ and the joint observation as $\mathrm{y}_{k,1:N_m}$, the joint likelihood can be either obtained as in [15] or by a formulation equivalent to the Coupled JPDA as

$$p(\mathrm{y}_{k,1:N_m}|\mathrm{x}_{k,1:N_t}) = \frac{\left[\prod_{j=1}^{N_m} \eta_c(\mathrm{y}_{k,j})\right]}{N_t!} \times \sum_{N_d=0}^{N_t} \frac{(\lambda_c V)^{N_m - N_d}\, e^{-\lambda_c V}}{(N_m - N_d)!} P_d^{N_d} \left(1 - P_d\right)^{N_t - N_d} \sum_{\phi_{k,1:N_t}|N_d} \prod_{i=1}^{N_t} \frac{p(\mathrm{y}_{k,\phi_{k,i}}|\mathrm{x}_{k,i})}{\eta_c(\mathrm{y}_{k,\phi_{k,i}})}. \quad (2.91)$$

The joint state vector $\mathrm{x}_{k,1:N_t} = (p_1, \ldots, p_{N_t}, v_1, \ldots, v_{N_t})_k^T$ is composed of position and velocity of all vehicles in the convoy, and the joint observation $\mathrm{y}_{k,1:N_m} = (\mathrm{y}_1, \ldots, \mathrm{y}_{N_m})_k^T$ contains position measurements of all targets and possible false alarms obtained at a given time instant k.

The joint bootstrap particle filter, the GPF and SDPF, and the SPF-GS consider their filtering densities to target a joint posterior density incorporating the joint multi-target likelihood function (2.91). Regarding the implementation of original particle flows, the GPF and SDPF reinterpret the filtered density empirically as a Gaussian pdf at the end of each iteration in order to avoid exponential growth of the number of mixture components over time. In contrast, this practical aspect does not affect the stochastic particle flow, whose filtered density is a mixture composed of a fixed number of local solutions to the actual posterior pdf.

Results

We simulated the trajectories of vehicles on a ring road by integrating the continuous-time stochastic IDM over 60 seconds with the parameters presented as follows. The convoy was set to start from rest with the vehicles initially positioned apart, led by a truck so that the queue of

cars is slowed down and forced into mutual proximity. The minimum allowed distance between vehicles was set to be exaggeratedly small ($s_0 = 0.5\,m$) to induce the model to control the distance between cars mainly based on the safe-time headway T_h. In this case, the safe-time headway indirectly determines the desired distance between vehicles, which is denoted as *target distance* in the table below. At the final steady state, the net speed of the convoy is dominated by the free-road speed of the truck, which motivates the safe-time headway being computed based on $v_{0,\text{truck}}$.

Parameter	**car**	**truck**
a	$0.5\,\text{m/s}^2$	$0.4\,\text{m/s}^2$
b	$1.5\,\text{m/s}^2$	$1.2\,\text{m/s}^2$
δ	4	4
v_0	$15\,\text{m/s}$	$10\,\text{m/s}$
s_0	$0.5\,\text{m}$	$0.5\,\text{m}$
T_h	$\frac{\{\text{target distance (m)}\}}{v_{0,\text{truck}}}$	-
l_α	$5\,\text{m}$	$20\,\text{m}$

Table 2.10: Parameters used for the IDM example

The joint state process covariance matrix is assumed as scaled by $\sigma_q^2 = 0.0625\ (\text{m/s})^2\,/\text{s}$, each position observation has variance $\sigma_r^2 = 4\,\text{m}^2$, $P_d = 0.80$, and $\lambda_c \times V = 0.01$ false alarm/scan, and the surveillance region's "volume" V, is, in fact, the length covered by a confidence region ($\approx 99.73\%$) that contains all the vehicles. Proposing a method to effectively initiate tracks was out of the example's scope, thus track initiation was considered to be ideal, i.e., the initial position and velocity of the targets are known with initial uncertainty scaled by the observation noise. The stochastic particle flow has been set with time horizon $T = 15\,\text{s}$ and integration step size $\Delta\lambda = 1\,\text{s}$.

Figure 2.20 shows two frames of an exemplary run, demonstrating the situation where a queue of cars is slowed down by a truck, forcing them into proximity. The non-filled rectangles depicted in Figure 2.20 denote the position estimates provided by the filter applied for that run. Interactions between vehicles in the convoy, due to proximity, can be well perceived by position estimates of the exemplary run as shown in Figure 2.19.

The resulting root-mean-square error (RMSE), normalized-estimation error squared (NEES), and average computation time (per time step) of all filters, for different numbers of vehicles, and target distance between vehicles $d = 10\,\text{m}$, are shown in Figures 2.16, 2.17 and 2.18 respectively. RMSE and NEES were computed over 100 Monte Carlo runs, with the particle-based filters using 200 samples. The following important aspects can be noted from Figure 2.16:

- In general, trackers that estimate on the joint $2N_t$-dimensional state space clearly outperform the classical multi-target trackers (JPDA, GNN, and JPDA-GM), both in terms of precision (RMSE) and credibility (NEES).

- Among the classical multi-target trackers, the Global Nearest Neighbor (GNN) association filter is the one that provides the most accurate estimates. This can be explained by an increasingly detrimental effect of the association uncertainty on estimation, which is more prominent in the JPDA and less prominent on the GNN filter.

- Estimation errors committed by the multi-target particle filter (joint SIR) grow exponentially with the number of state-space dimensions ($2N_t$), as expected, due to the curse of dimensionality.
- The Coupled PDA (CPDA) and the SPF-GS present commensurate root-mean-square errors, suggesting that most of their accuracy gain originates from tracking on the joint $2N_t$-dimensional state space and accounting for inherent dependencies between targets.
- The original particle flows (GPF, SDPF) show significant values of NEES. Most likely this is because the evaluated implementation of these filters cannot provide reliable estimates for state covariance matrices and depend on a dissimilar companion filter to work around it, which affects consistency of their estimates.
- The SPF-GS provides overall higher estimation accuracy (RMSE) and consistency (NEES), with low sensitivity to increasing the problem's number of dimensions.

Based on the results for this example, two important remarks are worth making. Firstly, the results show a notable performance improvement with adoption of filtering on the joint $2N_t$-dimensional state space: about 30-fold improvement in estimation precision (RMSE) and nearly 10-fold in credibility (NEES). The fact that the Coupled PDA performs as well as the SPF-GS suggests that modeling inherent dependencies between targets and filtering on the joint space provide most of the performance gain. Secondly, the example not only illustrates well the curse of dimensionality for the multi-target particle filter (joint SIR) but also corroborates the success of principled choices made in the SPF's formulation in order to avoid degeneracy in high-dimensional problems. This latter observation becomes clear when we realize that the performance indexes for the stochastic particle flow scale gently with the number of dimensions.

Additionally, it is also worth noting that the original particle flows provide relatively accurate estimates, scaling well with the number of dimensions. For the evaluated implementation, both particle flow filters (GPF, SDPF) rely on covariance matrices estimated by the CPDA as a companion filter. Due to this fact, calculated NEES for these filters is not reliable since their first and second-moment estimates are underpinned by distinct filtering methods. This does not disqualify the original particle filters per se since observed characteristics are probably due to the implementation settings. Under these circumstances, actual consistency (credibility) of their estimates cannot be quantified and, ultimately, evokes the question about the extent to which the success of the adopted implementation is due to the companion filter.

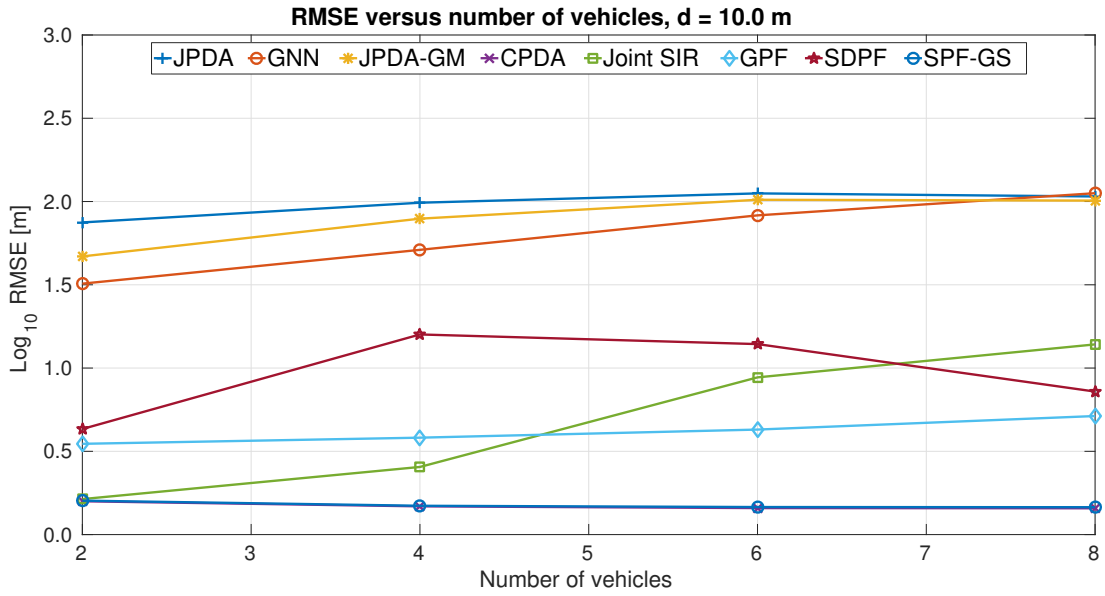


Figure 2.16: RMSE for the convoy tracking example

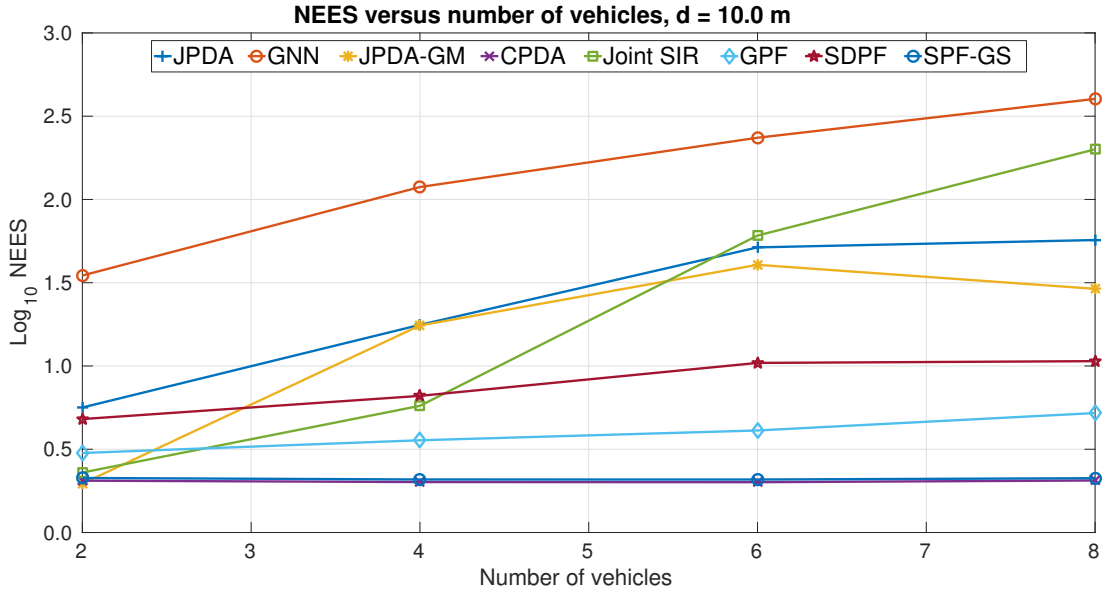


Figure 2.17: NEES for the convoy tracking example

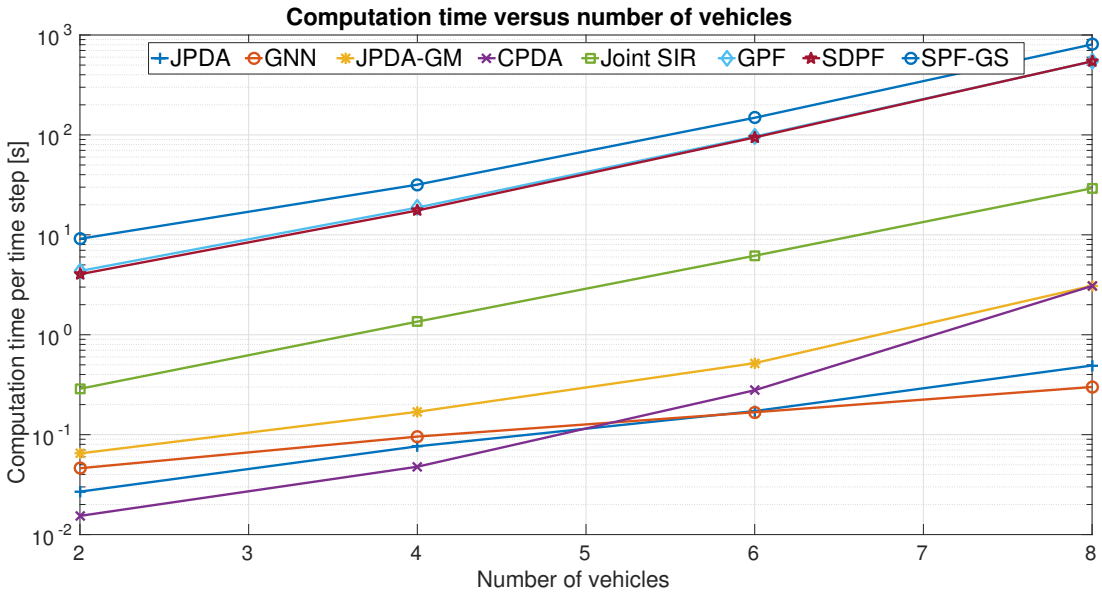


Figure 2.18: Average computation time for the convoy tracking example

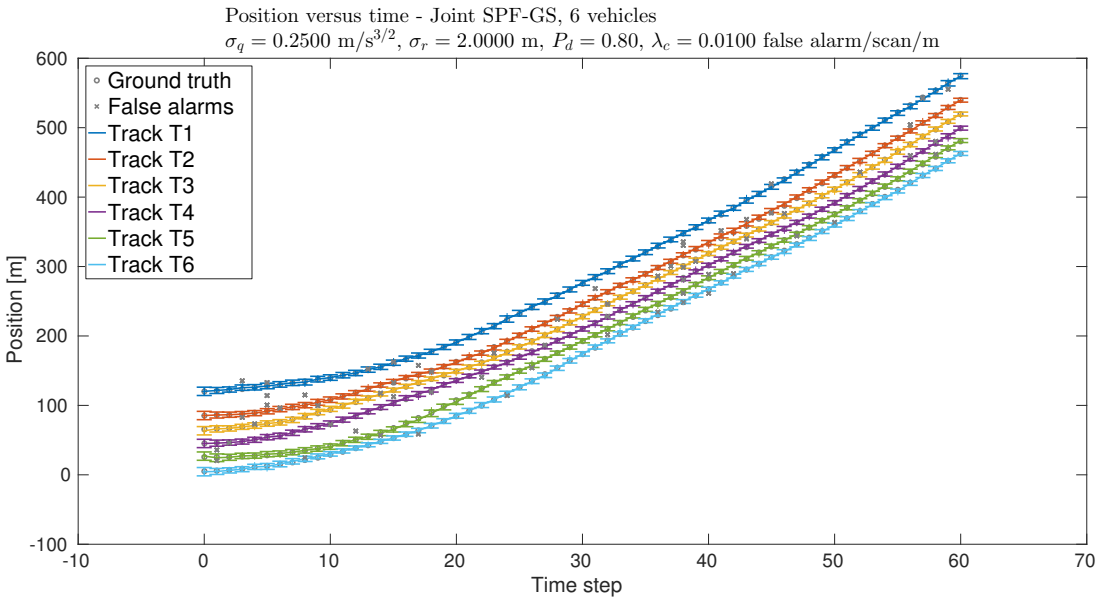
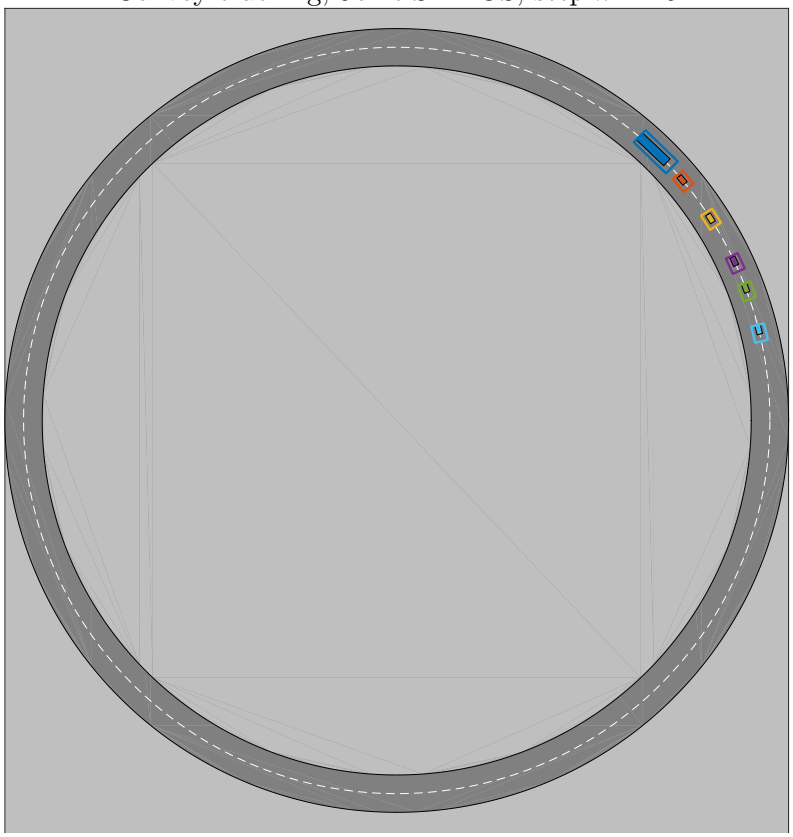


Figure 2.19: Position estimates for an exemplary run of the convoy tracking

Convoy tracking, Joint SPF-GS, step $k = 15$



Convoy tracking, Joint SPF-GS, step $k = 45$

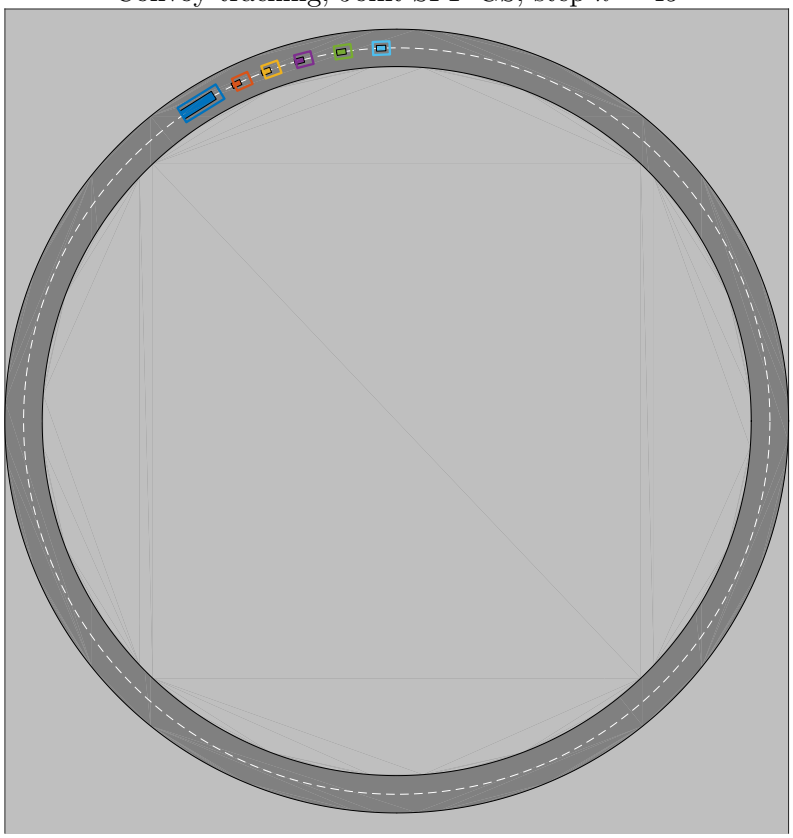


Figure 2.20: Illustration of convoy tracking on a ring road

2.7.4 Inference on Large Spatial Sensor Networks

In this section, we consider the problem proposed by Septier & Peters [175], posed to address inference of physical quantities of complex phenomena from a collection of noisy measurements obtained by a large network of spatially distributed sensors. According to Septier & Peters [175], a large number of applications could adopt such sensor networks to make inferences related to complex phenomena. Applications include environmental monitoring, weather forecasting etc. In this framework, a fusion center would regularly receive observations from sensors set up as a grid, which monitor a time-varying physical phenomenon presenting spatially diverse attributes such as pressure, temperature, concentrations of substance, radiation levels, seismic activity etc. Upon fusing the observations, the solution to the problem consists of estimating the phenomenon state at the current time instant at each of the sensor's positions. The problem becomes particularly challenging as the number of sensors in the grid increases, since solving the problem then demands efficient algorithms for inference in high-dimensions.

The physical phenomenon is modeled as a time-varying spatially-dependent continuous process defined over a two-dimensional space which is observed sequentially in time by a 2D spatial grid of $d_x = N_s$ sensors, where d_x is the state dimension. At time instant k, each sensor independently produces a noisy measurement of an attribute of interest about the phenomenon at its specific location, giving $\mathrm{y}_{k,j}|\mathrm{x}_k \sim p(\mathrm{y}_k^{(j)}|\mathrm{x}_k)$, $\forall j = 1, \ldots, N_s$. Based on the historic set of observations $\mathrm{Y}_{1:k} \coloneqq \{\mathrm{Y}_{k'} : k' = 1, \ldots, k\}$, where $\mathrm{Y}_k \coloneqq \{\mathrm{y}_{k,j} : j = 1, \ldots, N_s\}$, one is required to estimate, at time k, the state of the physical phenomenon $\mathrm{x}_k \in \mathbb{R}^{d_x}$ across the locations of all sensors in the grid. The state process that models the time-varying physical phenomenon is considered to follow a transition multivariate Generalized Hyperbolic (GH) density as

$$p_t\left(\mathrm{x}_k|\mathrm{x}_{k-1}\right) \propto \frac{\mathbb{K}_{c_1-d_x/2}\left[\sqrt{(c_2+Q(\mathrm{x}_k,\mathrm{x}_{k-1}))(c_3+\gamma^T\Sigma^{-1}\gamma)}\right]}{\left(\sqrt{(c_2+Q(\mathrm{x}_k,\mathrm{x}_{k-1}))(c_3+\gamma^T\Sigma^{-1}\gamma)}\right)^{d_x/2-c_1}} \cdot e^{(\mathrm{x}_k-\alpha\mathrm{x}_{k-1})\Sigma^{-1}\gamma} \tag{2.92}$$

where $Q(\mathrm{x}_k, \mathrm{x}_{k-1}) = (\mathrm{x}_k - \alpha\mathrm{x}_{k-1})^T\Sigma^{-1}(\mathrm{x}_k - \alpha\mathrm{x}_{k-1})$, $\alpha \in \mathbb{R}$ is the location constant, and $\mathbb{K}_{c_1}[\cdot]$ denotes the modified Bessel function of the second kind, of order c_1. The parameters c_1, c_2, and c_3 are scalar values that determine the shape of the distribution, $\Sigma \in \mathbb{R}^{d_x \times d_x}$ is the dispersion matrix, and the vector $\gamma \in \mathbb{R}^{d_x}$ is the skewness parameter. The choice of transition density in 2.92 can account for heavy-tailed and asymmetric data [175], which is beneficial when modeling physical process with extremal behavior. In special cases, the transition density becomes the normal, normal inverse Gaussian, skewed-t, and other densities. To generate the prior distribution at the first time step, we take $p_x(\mathrm{x}_0) = p_t(\mathrm{x}_0|\mathrm{x}_{-1} = 0)$.

The dispersion matrix is positive definite and is defined such that the degree of spatial correlation across sites of a physical phenomenon is given in terms of the separation between locations as

$$[\Sigma]_{ij} = \alpha_0 \exp\left[-\beta^{-1}\|\mathcal{S}_i - \mathcal{S}_j\|_2^2\right] + \alpha_1\delta_{ij}, \tag{2.93}$$

where $\|\cdot\|_2$ is the L2-norm, δ_{ij} the Kronecker symbol, $\alpha_0, \alpha_1 \in \mathbb{R}$, and $\mathcal{S}_m \in \mathbb{R}^2$ are the physical locations of the sensors for $m = 1, \ldots, N_s$.

For this example, we compare the performance of following filters:

- the Sequential Importance Resampling filter (SIR);

- the block SIR filter, which partitions the state space into separate subspaces of smaller dimensions (blocks of 4 sensors each) and run a particle filter on each subspace [160];
- the Sequential manifold Metropolis-Adjusted Algorithm (SmMALA) filter [80, 175];
- the Sequential manifold Hamiltonian Monte Carlo (SmHMC) filter [80, 175];
- the Stochastic Particle Flow, Gaussian sum (SPF-GS).

These filters are compared for two cases:

- a Gaussian state process and a Gaussian likelihood;
- a Skewed-t state process and Poisson-distributed observations.

Note that SmMALA and SmHMC are chosen because we perceive they constitute two of the best sequential MCMC filters that exist. It is essential to justify why we have not included annealed importance sampling (AIS) [147] and SMC samplers [56] in our comparisons. Although these techniques are built on fast mixing Markov chains, their filtering procedures operate on the joint space along the complete path of samples. This makes them highly prone to the curse of dimensionality for long-time horizons. A careful explanation of this issue can be found in [110] (and Section 2.6.1) but the key point is that as dimension increases, it becomes increasingly important to avoid consideration of the path. SMC samplers could be adapted to filter in the marginal space (at the cost of up to $\mathcal{O}(N^3)$ evaluations) but this would require an ad-hoc approximation to the target pdf. Developing the approximations that would be needed to enable SMC samplers to be applied to the problem we consider is not the focus of the chapter. We have compared performance of SPF-GS with state-of-the-art techniques (SmMALA and SmHMC) that consider the marginal distribution as well as techniques (SIR and block SIR) that we perceive to be good examples of the class of algorithms, which also includes AIS and SMC samplers, that consider the joint space of the complete sample path and can be applied, without modifications, to the problem we are focused on solving.

These filters are compared for two cases:

- Gaussian state process and Gaussian likelihood;
- Skewed-t state process and Poisson-distributed observations.

The implementations of SmMALA and SmHMC used are exactly as made available by Septier & Peters [175]. These algorithms make use of a refinement step of the state [175] at the current time, performed with a random partitioning of size 4, by using the empirical approximation of the previous posterior distribution as proposal distribution.

Results

Gaussian State Process and Gaussian Likelihood

We first consider a trivial special case of the GH family as the transition density, namely the multivariate normal distribution. In this setting, each sensor measures the attribute of a physical process with some Gaussian noise. The resulting model is given by

$$\begin{aligned} p_t(\mathrm{x}_k|\mathrm{x}_{k-1}) &= \mathcal{N}(\mathrm{x}_k;\, \alpha\mathrm{x}_{k-1}, \Sigma), \\ p(\mathrm{y}_k|\mathrm{x}_k) &= \mathcal{N}(\mathrm{y}_k;\quad \mathrm{x}_k,\quad \mathrm{R}), \end{aligned} \tag{2.94}$$

where $\mathrm{R} = \sigma_y^2 \mathbb{I}_{d_x}$, and the following model parameters are used: $\alpha = 0.9$, $\sigma_y^2 = 2$, and with the dispersion matrix constructed using $\alpha_0 = 3$, $\alpha_1 = 0.01$, $\beta = 20$. When performing the comparison between the filters, we use as a reference the estimates provided by the Kalman filter (which is optimal in this special case). For the SmMALA and SmHMC, the proposed metric tensor is given by $\mathrm{G}(\mathrm{x}_k) = \mathrm{R}^{-1} + \Sigma^{-1}$.

The methodology presented in [175] for performance evaluation was reproduced. Instead of presenting the baseline (provided by the Kalman filter) performance explicitly, the accuracy of each filter is evaluated with respect to the baseline performance, i.e. we observe the difference between the log mean square error for each filter against the log mean square error for the Kalman filter. The relative log mean square error, log normalized-estimation error squared (NEES) and average computation time (per time step) of all filters, for different numbers of sensors in the grid, are shown in Figures 2.21, 2.22, 2.23 respectively. The mean square error of the estimates were computed over 100 Monte Carlo runs, with the particle-based filters using 200 samples. The step size adopted for the sequential MCMC filters is $\Delta\lambda = 0.5$ and the number of steps obtained as $L = L_0 + N$, where $L_0 = 0.2N$ is the number of steps for the burn-in phase, whereas for the SPF-GS we applied the empirical rules for time horizon and step step size as presented in subsection 2.4.6. From Figures 2.21, 2.22, 2.23, we can note that for this example:

- The performances of all sequential MCMC filters are in accordance with the results shown in [175].
- The SPF-GS outperforms all other filters in terms of mean square error and normalized-estimation error squared.
- The SPF-GS demands the highest computational effort when the number of sensors is small, and its computing time scales better than that of SmMALA and worse than that of SmHMC for higher dimensions.

The results suggest that SPF-GS is the most accurate among the compared methods. It is unambiguous that the consistency of the estimates produced by the SPF-GS is better than that by all other filters: the NEES for SPF-GS is very close to one (from above) for all evaluated dimensions. It may be well possible that better results could be achieved for the sequential MCMC filters by carefully choosing the step size and number of steps. This would slightly change the performance indexes. However, we strongly believe that such changes would not be enough to modify the conclusions. Additional tests with the SPF-GS demonstrate the computational cost can be directly traded with estimation accuracy. By allowing more steps, via the criteria of Section 2.4.6, SPF-GS is very close to optimal as depicted in Figure 2.21. On the other hand, if one fixes the step size to $\Delta\lambda = 0.5$ and number of steps to $L = 20$, the results are as presented in Figures 2.21 and 2.23, where it becomes clear the computational cost alleviation at the expense of slightly degrading mean square error.

Skewed-t State Process with Poisson-Distributed Observations

A high-dimensional non-linear and non-Gaussian state-space model is now studied. The transition kernel is proposed to be a multivariate GH skewed-t density described by (2.92) with

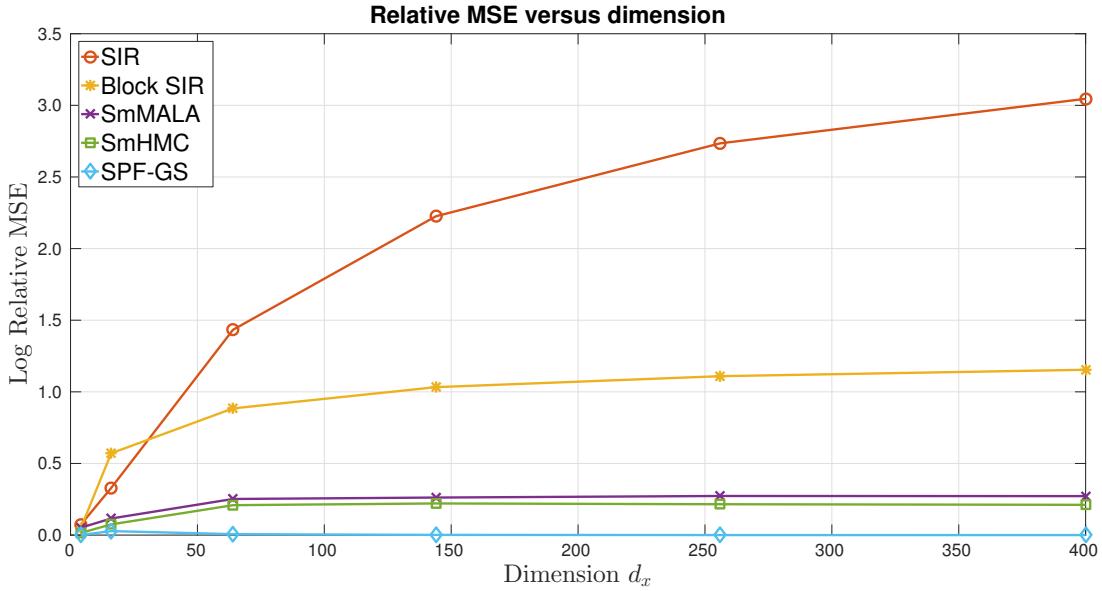


Figure 2.21: Relative MSE for the linear, Gaussian sensor network example

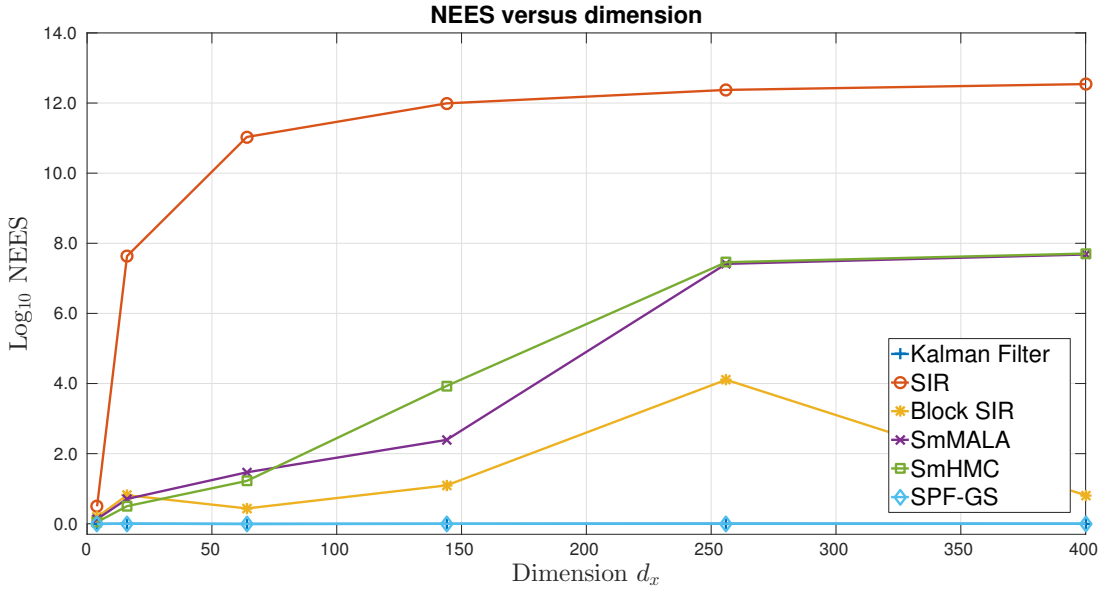


Figure 2.22: NEES for the linear, Gaussian sensor network example

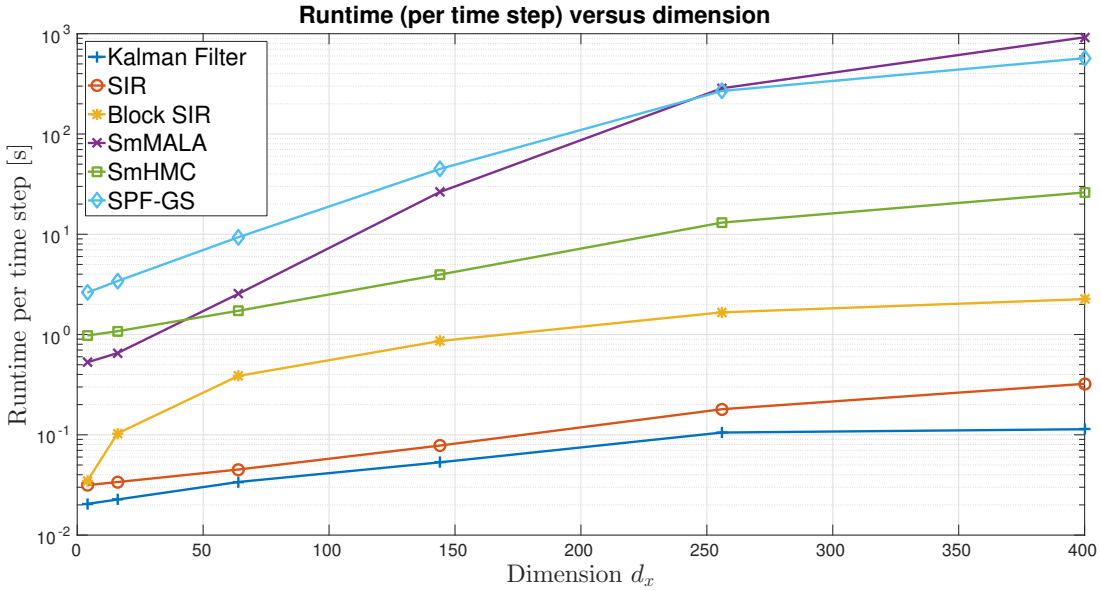


Figure 2.23: Average runtime for the linear, Gaussian sensor network example

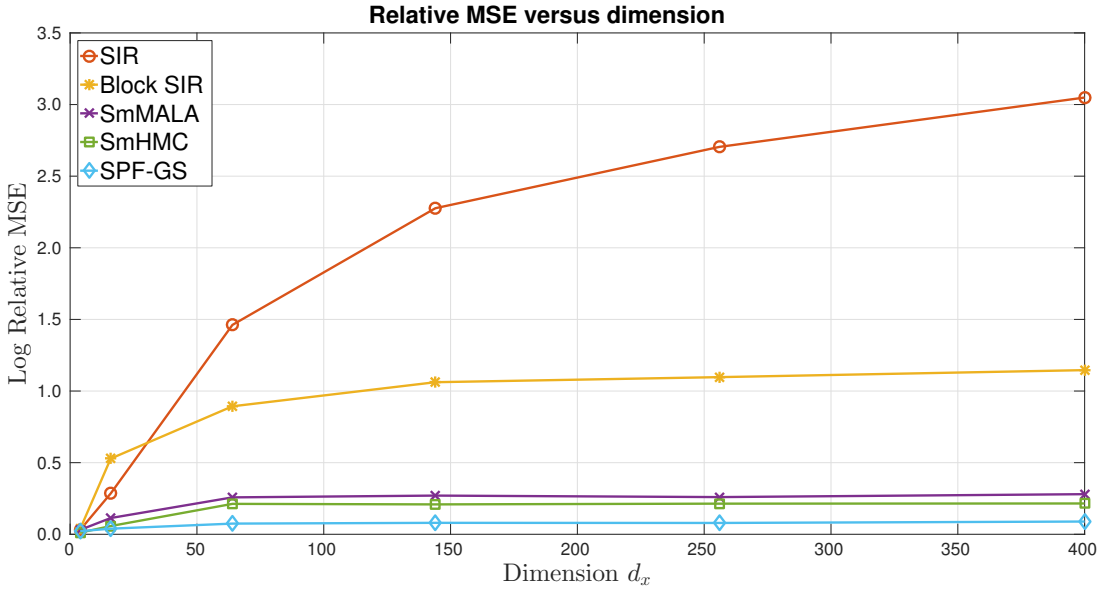


Figure 2.24: Relative MSE for the linear, Gaussian example (SPF-GS with $\Delta\lambda = 0.5$, $L = 20$)

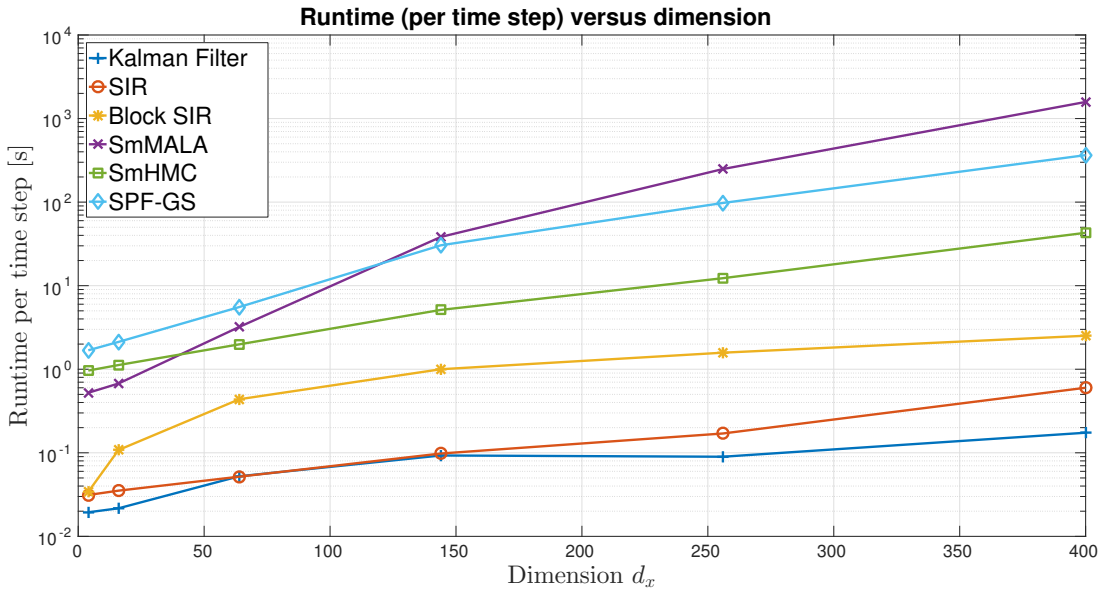


Figure 2.25: Runtime for the linear, Gaussian example (SPF-GS with $\Delta\lambda = 0.5$, $L = 20$)

$c_1 = -\nu/2$, $c_2 = \nu$ and $c_3 = 0$. The likelihood function is assumed to be a Poisson distribution, highly non-linear on the state x_k, given by

$$p(\mathrm{y}_k|\mathrm{x}_k) = \prod_{j=1}^{d_x} \frac{\lambda_j(\mathrm{x}_{j,k})^{\mathrm{y}_{j,k}}}{\mathrm{y}_{j,k}!} e^{-\lambda_j(\mathrm{x}_{j,k})}, \quad \lambda_j(\mathrm{x}_{j,k}) = m_1 e^{m_2 \mathrm{x}_{j,k}}, \tag{2.95}$$

such that $\mathrm{x}_k = (\mathrm{x}_{1,k}, \ldots, \mathrm{x}_{d_x,k})^T$ and $\mathrm{y}_k = (\mathrm{y}_{1,k}, \ldots, \mathrm{y}_{d_x,k})^T$. The model parameters are fixed as $m_1 = 1$, $m_2 = 1/3$, $\alpha = 0.9$, $\nu = 7$, $\gamma = 0.3\mathbb{1}_{d_x \times 1}$ with the dispersion matrix constructed using $\alpha_0 = 3$, $\alpha_1 = 0.01$ and $\beta = 20$. The implied prior density is not log concave, and thus the tensor metric that defines the diffusion coefficient for the sequential MCMC algorithms is modified according to

$$\mathrm{G}(\mathrm{x}_k) = \Lambda(\mathrm{x}_k) + \tilde{\Sigma}^{-1}, \tag{2.96}$$

where

$$\Lambda(\mathrm{x}_k) = m_1 m_2^2 \begin{pmatrix} e^{m_2 \mathrm{x}_{1,k}} & & 0 \\ & \ddots & \\ 0 & & e^{m_2 \mathrm{x}_{d_x,k}} \end{pmatrix}, \tag{2.97}$$

$$\tilde{\Sigma} = \frac{\nu}{\nu - 2}\Sigma + \frac{\nu^2}{(2\nu - 8)(1/2 - 1)^2}\gamma\gamma^{\mathrm{T}}, \quad \nu > 4. \tag{2.98}$$

For this problem, the local linearization of the stochastic particle flow around a probability mass x_l, analogous to (2.55) and (2.56), is given by

$$\frac{1}{2}\mathrm{D}\nabla_{\mathrm{x}} \log \tilde{\pi}(\mathrm{x}) \approx \mathrm{C}(\mathrm{x}_l) \cdot \mathrm{x} + \mathrm{c}(\mathrm{x}_l),$$

$$\mathrm{C}(\mathrm{x}_l) = \frac{1}{2}\mathrm{D}\left(-\mathrm{V}(\mathrm{x}_l)^{-1} - \mathrm{P}_{k|k-1}^{-1}\right), \tag{2.99}$$

$$\mathrm{c}(\mathrm{x}_l) = \frac{1}{2}\mathrm{D}\left(m_2 \mathrm{y}_k - \mathrm{v}(\mathrm{x}_l) + \mathrm{V}(\mathrm{x}_l)^{-1}\mathrm{x}_l + \mathrm{P}_{k|k-1}^{-1}\alpha\mu_{m,k-1}\right), \tag{2.100}$$

where $\mathrm{V}(\mathrm{x}_l)^{-1} = \Lambda(\mathrm{x}_l)$, $\mathrm{v}(\mathrm{x}_l) = m_1 m_2 e^{m_2 \mathrm{x}_l}$, and

$$\mathrm{P}_{k|k-1} = \mathbb{E}\left[(\mathrm{x}_{k|k-1} - \alpha\mu_{m,k-1})(\mathrm{x}_{k|k-1} - \alpha\mu_{m,k-1})^T\right]. \tag{2.101}$$

Once again, we follow the methodology presented in [175] for performance evaluation. The log root-mean-square error, log normalized-estimation error squared (NEES) and average computation time (per time step) of all filters, for different numbers of sensors in the grid, are shown in Figures 2.26, 2.27, and 2.28 respectively. The mean square error of the estimates were computed over 100 Monte Carlo runs, with the particle-based filters using 200 samples. The step size adopted for the sequential MCMC filters is $\Delta\lambda = 0.5$ and the number of steps computed as $L = L_0 + N$, where $L_0 = 0.2N$ is the number of steps for the burn-in phase, whereas for the SPF-GS we applied the empirical rules for time horizon and step-step size as presented in Subsection 2.4.6. In Figure 2.29 we illustrate the posterior means and variances of the state across the sensors grid ($d_x = 400$) at different time steps, for all evaluated filters. From Figures 2.26, 2.27, and 2.28, we can note that

- The performances of all sequential MCMC filters are in accordance with the results shown in [175].
- The SPF-GS presents performance commensurate to that of the SmHMC filter in terms of root-mean-square error, outperforming all other filters.
- The SPF-GS outperforms all other filters in terms normalized-estimation error squared.
- The SPF-GS demands a computational effort higher than that of SmMALA when the number of sensors is small but its computing time scales better than that of SmMALA and similarly to that of SmHMC for higher dimensions.

For this example, the results indicate that the SPF-GS is as accurate as the SmHMC filter in general, and that SPF-GS is the most accurate method in high dimensional problems. Once again, it is clear that the consistency (credibility) of estimates by the SPF-GS is higher than that by all other filters: the NEES is very close to one (from above) for all evaluated dimensions. It is worth noting that, to keep the computational cost for the SPF-GS competitive, the total number of steps was limited. This did result in some loss of accuracy but the implication is perhaps that improved performance is possible if sufficient computational resources are available. The associated trade-off is sufficiently complex to form a hard obstacle against systematic solutions. However, the techniques presented in Section 2.4.6 proved sufficient to generate the results presented here.

2.8 Conclusions

This chapter focused on a crucial issue of multi-object estimators touched on in Chapter 0: describing the filtering probability distributions of high-dimensional nonlinear phenomena to an adequate level of accuracy despite the curse of dimensionality and irregularities caused by usual Monte Carlo approximations.

The chapter builds on concepts such as continuous-time filtering and sequential Monte Carlo methods, being intensely studied by the research community. It provides a description of

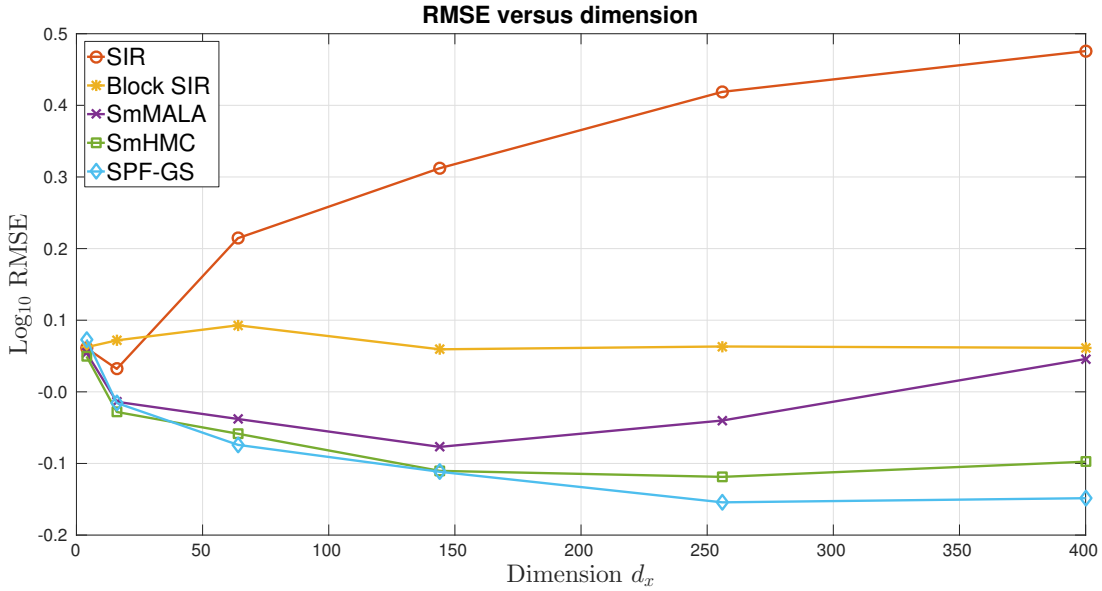


Figure 2.26: RMSE for the nonlinear, non-Gaussian sensor network example

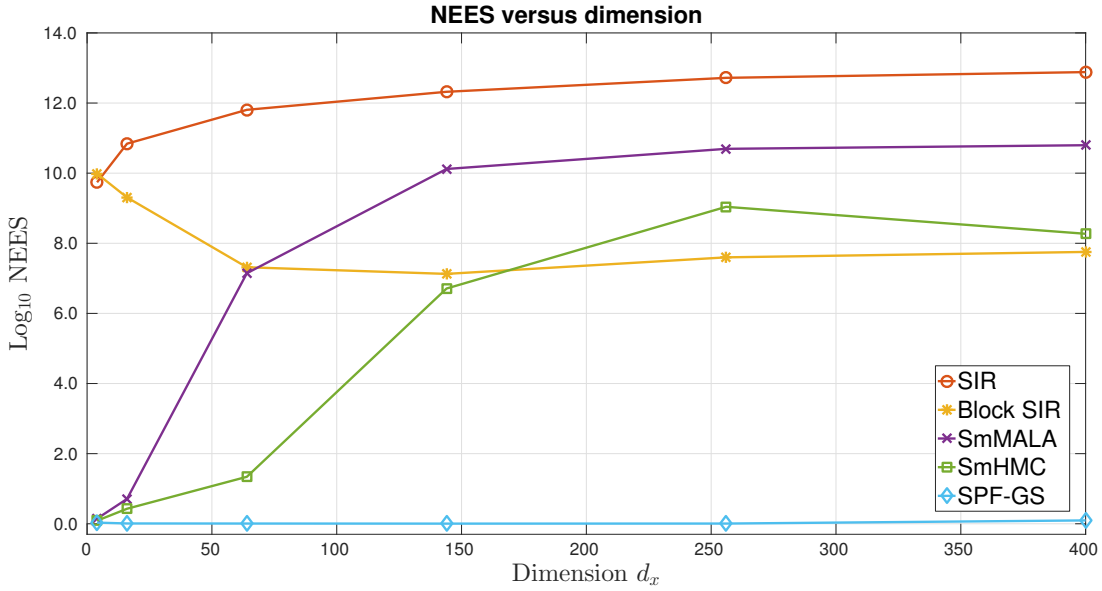


Figure 2.27: NEES for the nonlinear, non-Gaussian sensor network example

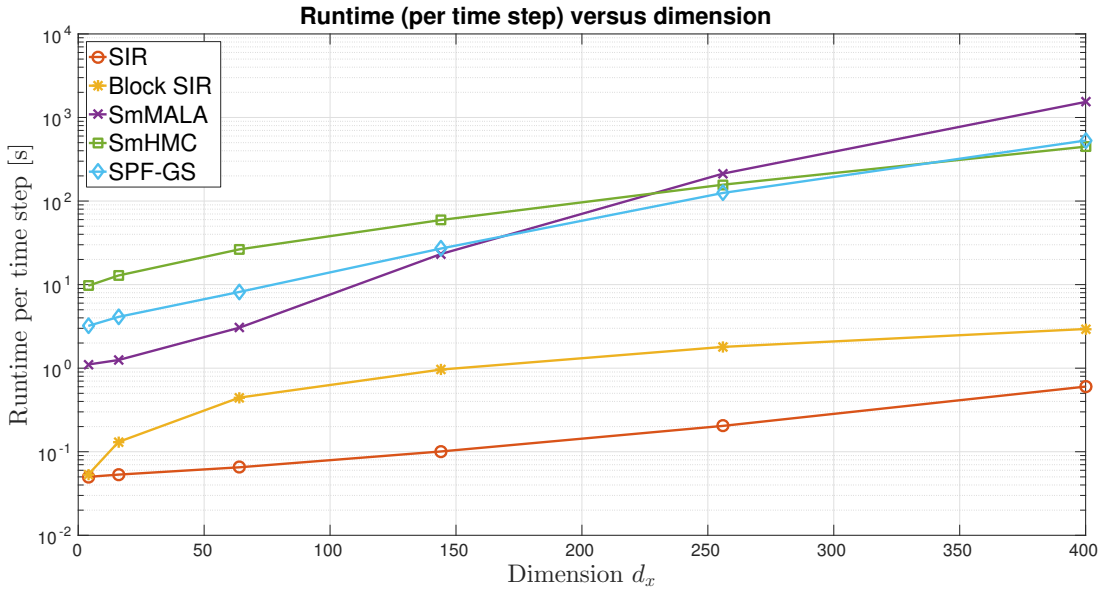


Figure 2.28: Average runtime for the nonlinear, non-Gaussian sensor network example

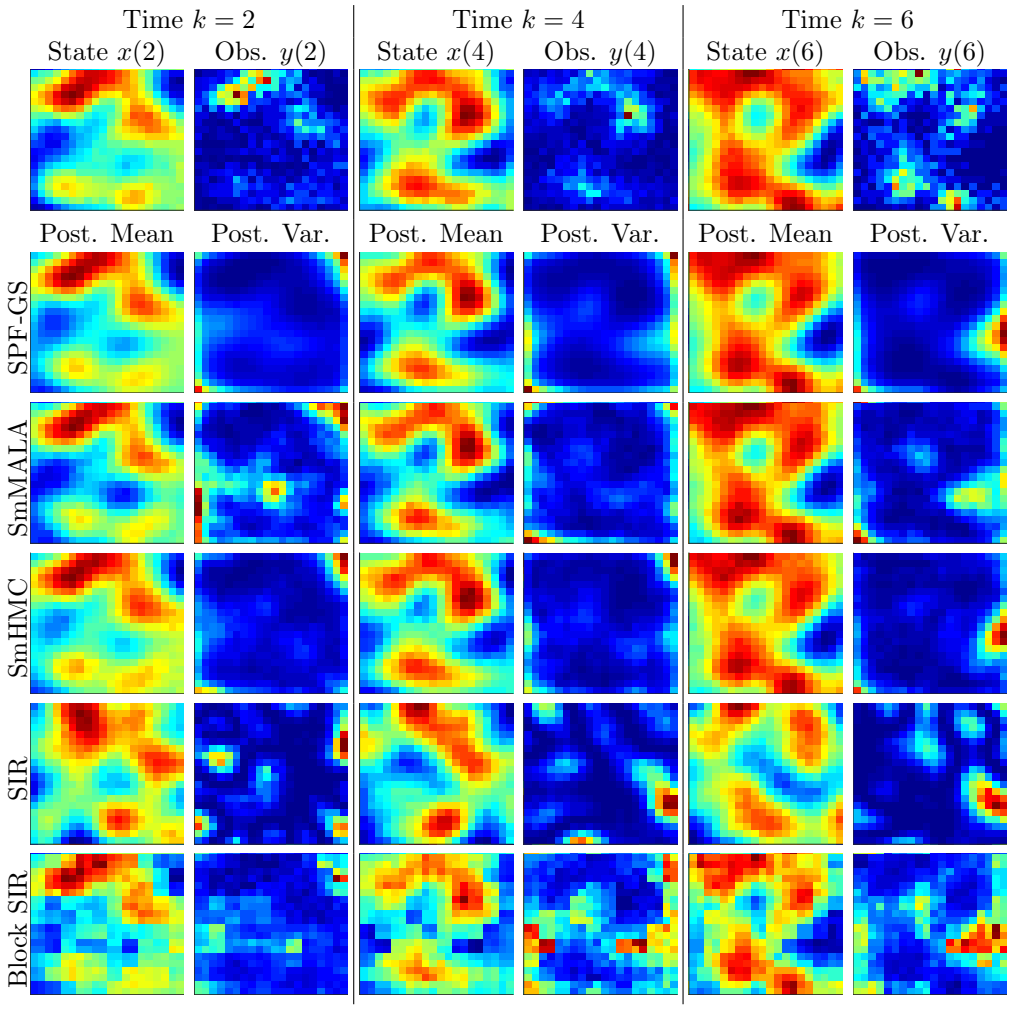


Figure 2.29: Posterior statistics for the nonlinear, non-Gaussian sensor example ($d_x = 400$)

these concepts that aims to draw out some key insights from the theoretical research into sequential Monte Carlo filtering while also responding to the significant empirical challenges encountered when applying existing and emerging tools to difficult real-world problems. More specifically, the chapter is part of a growing body of research which aims to apply concepts from the sequential Monte Carlo community to problems involving high-dimensional spaces. This broader body of research and this chapter, in particular, is motivated by the increasing demand for more statistically efficient methods to solve difficult inference problems exemplified by those involving large numbers of dimensions and relevant to a vast range of applications.

Also, the chapter capitalizes on some important findings that have been reported recently [198, 160, 175] regarding how the local properties of sequential Monte Carlo filtering measures impact algorithms' abilities to solve high-dimensional problems. We exploit the observation made by Rebeschini & van Handel [160], that, by using the decay of correlations property, it is possible to develop particle filters based on local solutions whose approximation error becomes less sensitive to augmenting the number of state dimensions.

Within this context, we proposed a novel filter which aims to address the well-known shortcomings of sequential Monte Carlo methods when applied to nonlinear high-dimensional filtering problems. The novel method uses a Monte Carlo procedure to generate a sequence of equally-weighted samples that each guide a local solution of the Fökker-Planck equation. Using these local approximations, a mixture is produced that approximates the filtering density. The result is a statistically-sound general-purpose class of algorithms. In the context of a simple, though not trivial, high-dimensional inference problem and in comparison with state-of-the-art algorithms, the proposed approach has been shown to offer significant improvement in statistical consistency with a commensurate computational expense.

In its most computationally efficient form, SPF-GS, the proposed filter has a complexity bounded by only $\mathcal{O}\left(NLd_x^3\right)$ evaluations[13] and it has the appealing property that its operations per sample (and the associated mixture component) can be parallelized. When articulated as a marginal particle filter, SPF-MPF, the complexity is bounded by $NLd_x^3 + N + N^2 \sim \mathcal{O}(NLd_x^3)$ evaluations. That said, our investigations indicate that further consideration is required in order to explore fully the potential of the stochastic particle flow. Future work will focus on the computational cost of the algorithms and on guaranteeing certain properties of the diffusion matrix, (e.g., ensuring the matrix is positive definite and not singular). It should be possible, at least in the context of some statistical models, to reduce the computational complexity by exploiting or imposing sparsity in the diffusion matrix. Girolami & Calderhead [80] suggest that the use of guiding Hamiltonians [67] could provide a way of reducing such computational cost but it is currently unclear how such a solution would be adopted in the context of the stochastic particle flow. Another potentially promising future direction would be adopting the same approach as the well-known Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm and thereby work around the need to explicitly evaluate the Hessian matrix at all simulation steps. We also strongly believe that it would be possible for future work to result in further improvements on the bounds used to compute the step size and number of simulation steps. Such advances will likely improve the computational efficiency of the algorithm and are the subject of ongoing development.

[13] It is important to remark that the stochastic particle flow is much more computationally complex than the original particle flows, which are generally bounded by $\mathcal{O}(N)$ computations both in theory and practice.

2.9 Proofs and Derivations

2.9.1 Bounds for the Time Horizon and Step Size

General Assumptions

Let $\Phi : \mathbb{R}^{d_x} \to \mathbb{R}$ be a measurable convex function, satisfying

$$\int_{\mathbb{R}^{d_x}} \exp\{-\Phi(\mathrm{x})\} < \infty, \tag{2.102}$$

$$\Phi(\mathrm{x}) - \Phi(\bar{\mathrm{x}}) - \nabla_{\mathrm{x}}\Phi(\bar{\mathrm{x}})^T(\mathrm{x}-\bar{\mathrm{x}}) \geq \frac{1}{2}m\left\|\mathrm{x}-\bar{\mathrm{x}}\right\|_2^2, \tag{2.103}$$

$$\left\|\nabla_{\mathrm{x}}\Phi(\mathrm{x}) - \nabla_{\mathrm{x}}\Phi(\bar{\mathrm{x}})\right\|_2 \leq M\left\|\mathrm{x}-\bar{\mathrm{x}}\right\|_2, \quad \forall \mathrm{x}, \bar{\mathrm{x}} \in \mathbb{R}^{d_x}, \tag{2.104}$$

for two existing positive constants m and M. Let $\bar{\mathrm{x}} \in \mathbb{R}^{d_x}$ be the global minimum of $\Phi(\mathrm{x})$. We define the log-concave target density for a Langevin algorithm as $\pi(\mathrm{x}) = e^{-\Phi(\mathrm{x})}\left(\int_{\mathbb{R}^{d_x}} e^{-\Phi(\mathrm{x})}d\mathrm{x}\right)^{-1}$.

Lemma 2.3. *Suppose a probability measure $\mathcal{P}_{\mathcal{L},T}$ produced by the exact integration, up to time horizon T, of the Langevin diffusion SDE*

$$d\mathcal{L}_\lambda = -\frac{1}{2}\mathrm{D}_\lambda\nabla\Phi(\mathcal{L}_\lambda)d\lambda + \mathrm{D}_\lambda^{1/2}d\mathrm{W}_\lambda,\ \lambda \geq 0,\ \mathcal{L}_0 = 0, \tag{2.105}$$

departing from the initial density $\nu(\mathrm{x})$ and targeting the invariant density $\pi(\mathrm{x}) \propto \exp\{-\Phi(\mathrm{x})\}$ with measure $\mathcal{P}_\pi(d\mathrm{x})$. Process $\{\mathrm{W}_\lambda\}_{\lambda\geq 0}$ is the standard Wiener process and D_λ is the diffusion matrix. Under assumptions (2.102), (2.103), and (2.104),

$$\left\|\mathcal{P}_{\mathcal{L},T} - \mathcal{P}_\pi\right\|_{TV} \leq e^{-\frac{1}{2}Tm}\mathbb{E}^{\mathcal{P}_\pi}\left[\nu^2/\pi^2\right]^{1/2},\ T \geq 0. \tag{2.106}$$

Proof. By denoting the Markov transition kernel as $P_t(\mathrm{x},\cdot)$, an argument similar to that given by Dalalyan [41], invoking the Cauchy-Schwarz inequality and the spectral gap bound for the transition operator, gives

$$\begin{aligned}
\left\|\mathcal{P}_{\mathcal{L},T} - \mathcal{P}_\pi\right\|_{\mathrm{TV}} &= \sup_{A\in\mathscr{B}(\mathbb{R}^{d_x})}\left|\int_{\mathbb{R}^{d_x}} P_t(\mathrm{x},A)\nu(\mathrm{x})d\mathrm{x} - \mathcal{P}_\pi(A)\right| \\
&= \sup_{A\in\mathscr{B}(\mathbb{R}^{d_x})}\left|\int_{\mathbb{R}^{d_x}} \left(P_t(\mathrm{x},A) - \mathcal{P}_\pi(A)\right)\nu(\mathrm{x})d\mathrm{x}\right| \\
&= \sup_{A\in\mathscr{B}(\mathbb{R}^{d_x})}\left|\int_{\mathbb{R}^{d_x}} \left(P_t(\mathrm{x},A) - \mathcal{P}_\pi(A)\right)\frac{\nu(\mathrm{x})}{\pi(\mathrm{x})}\pi(\mathrm{x})d\mathrm{x}\right| \\
&\leq \sup_{A\in\mathscr{B}(\mathbb{R}^{d_x})}\int_{\mathbb{R}^{d_x}} \left|P_t(\mathrm{x},A) - \mathcal{P}_\pi(A)\right|\left|\frac{\nu(\mathrm{x})}{\pi(\mathrm{x})}\right|\pi(\mathrm{x})d\mathrm{x} \\
&\leq \sup_{A\in\mathscr{B}(\mathbb{R}^{d_x})}\left[\int_{\mathbb{R}^{d_x}} \left|P_t(\mathrm{x},A) - \mathcal{P}_\pi(A)\right|^2\pi(\mathrm{x})d\mathrm{x}\right]^{\frac{1}{2}}\left(\int_{\mathbb{R}^{d_x}} \left|\nu(\mathrm{x})/\pi(\mathrm{x})\right|^2\pi(\mathrm{x})d\mathrm{x}\right)^{\frac{1}{2}} \\
\text{(Dalalyan, 2014)} &\leq \frac{1}{2}e^{-\frac{1}{2}mT}\mathbb{E}^{\mathcal{P}_\pi}\left[\nu^2/\pi^2\right]^{1/2},
\end{aligned} \tag{2.107}$$

where $\mathscr{B}(\mathbb{R}^{d_x})$ is the σ-algebra composed of the Borel sets of $\mathbb{R}^{d_x}$. □

Lemma 2.4. *Under conditions (2.102), (2.103) and (2.104), and assumptions of Lemma 2.3, given the initial probability density* $\nu(\mathrm{x}) = \mathcal{N}(\mathrm{x};\, \mu_\nu, \sigma_\nu^2 \mathbb{I}_{d_x})$*, for* $\sigma_\nu^2 < 2M^{-1}$*, then*

$$\begin{aligned}&\|\mathcal{P}_{\mathcal{L},T} - \mathcal{P}_\pi\|_{TV} \\ &\qquad \leq \frac{1}{2}\exp\left\{-\frac{1}{2}mT + \frac{d_x}{4}\log\left(\frac{1}{m\sigma_\nu^2(2-M\sigma_\nu^2)}\right) + \frac{M}{2(2-M\sigma_\nu^2)}\|\bar{\mathrm{x}} - \mu_\nu\|_2^2\right\}. \quad (2.108)\end{aligned}$$

Proof. According to *Lemma 4* in [41], if (2.104) holds then

$$\Phi(\mathrm{x}) - \Phi(\bar{\mathrm{x}}) - \nabla_{\mathrm{x}}\Phi(\bar{\mathrm{x}})^T(\mathrm{x} - \bar{\mathrm{x}}) \leq \frac{M}{2}\|\mathrm{x} - \bar{\mathrm{x}}\|_2^2. \quad (2.109)$$

Thus,

$$\begin{aligned}\pi(\mathrm{x}) &\geq e^{-\frac{M}{2}\|\mathrm{x}-\bar{\mathrm{x}}\|_2^2 - \Phi(\bar{\mathrm{x}})}\left(\int_{\mathbb{R}^{d_x}} e^{-\frac{m}{2}\|\mathrm{x}-\bar{\mathrm{x}}\|_2^2 - \Phi(\bar{\mathrm{x}})}d\mathrm{x}\right)^{-1} \\ &= e^{-\frac{M}{2}\|\mathrm{x}-\bar{\mathrm{x}}\|_2^2 - \Phi(\bar{\mathrm{x}}) + \Phi(\bar{\mathrm{x}})}\left(\int_{\mathbb{R}^{d_x}} e^{-\frac{m}{2}\|\mathrm{x}-\bar{\mathrm{x}}\|_2^2}d\mathrm{x}\right)^{-1}, \\ \pi(\mathrm{x}) &\geq \left(2\pi m^{-1}\right)^{-d_x/2} e^{-\frac{M}{2}\|\mathrm{x}-\bar{\mathrm{x}}\|_2^2}. \end{aligned} \quad (2.110)$$

We use (2.110), $\sigma_\nu^{-2} > M/2$, and define $W = 2\sigma_\nu^{-2}$ to compute

$$\begin{aligned}&\mathbb{E}^{\mathcal{P}_\pi}\left[\nu^2/\pi^2\right] = \int_{\mathbb{R}^{d_x}}\left(\frac{\nu}{\pi}\right)^2\pi(\mathrm{x})d\mathrm{x} = \int_{\mathbb{R}^{d_x}}\frac{\nu^2}{\pi}d\mathrm{x} \\ &\leq \int_{\mathbb{R}^{d_x}}\frac{\left[\left(2\pi\sigma_\nu^2\right)^{-d_x/2} e^{-\frac{\sigma_\nu^{-2}}{2}\|\mathrm{x}-\mu_\nu\|_2^2}\right]^2}{\left(2\pi m^{-1}\right)^{-d_x/2} e^{-\frac{M}{2}\|\mathrm{x}-\bar{\mathrm{x}}\|_2^2}}d\mathrm{x} \\ &= \left(2\pi\sigma_\nu^2\right)^{-d_x}\left(2\pi m^{-1}\right)^{d_x/2}\int_{\mathbb{R}^{d_x}}\frac{e^{-\sigma_\nu^{-2}\|\mathrm{x}-\mu_\nu\|_2^2}}{e^{-\frac{M}{2}\|\mathrm{x}-\bar{\mathrm{x}}\|_2^2}}d\mathrm{x} \\ &= \left(2\pi\cdot 2W^{-1}\right)^{-d_x}\left(2\pi m^{-1}\right)^{d_x/2}\int_{\mathbb{R}^{d_x}} e^{-\frac{W}{2}\|\mathrm{x}-\mu_\nu\|_2^2 + \frac{M}{2}\|\mathrm{x}-\bar{\mathrm{x}}\|_2^2}d\mathrm{x} \\ &= \frac{\left(2\pi m^{-1}\right)^{d_x/2}}{\left(2\pi\cdot 2W^{-1}\right)^{d_x}} e^{\frac{1}{2}(W-M)^{-1}WM\|\bar{\mathrm{x}}-\mu_\nu\|_2^2}\int_{\mathbb{R}^{d_x}} e^{-\frac{(W-M)}{2}\|\mathrm{x}-(W-M)^{-1}(W\mu_\nu - M\bar{\mathrm{x}})\|_2^2}d\mathrm{x} \\ &= \frac{\left(2\pi m^{-1}\right)^{d_x/2}}{\left(2\pi\cdot 2W^{-1}\right)^{d_x}} e^{\frac{1}{2}\left(M^{-1}-W^{-1}\right)^{-1}\|\bar{\mathrm{x}}-\mu_\nu\|_2^2}\left(2\pi(W-M)^{-1}\right)^{d_x/2} \\ &= \left(\frac{W^2}{4m(W-M)}\right)^{d_x/2} e^{\frac{1}{2}\left(M^{-1}-W^{-1}\right)^{-1}\|\bar{\mathrm{x}}-\mu_\nu\|_2^2} \\ &= \left(\frac{\sigma_\nu^{-4}}{m(2\sigma_\nu^{-2}-M)}\right)^{d_x/2} e^{\frac{1}{2}\left(M^{-1}-\frac{1}{2}\sigma_\nu^2\right)^{-1}\|\bar{\mathrm{x}}-\mu_\nu\|_2^2} \\ &= \left(\frac{1}{m\sigma_\nu^2(2-M\sigma_\nu^2)}\right)^{d_x/2} e^{M\left(2-M\sigma_\nu^2\right)^{-1}\|\bar{\mathrm{x}}-\mu_\nu\|_2^2} \\ &\therefore \mathbb{E}^{\mathcal{P}_\pi}\left[\nu^2/\pi^2\right]^{1/2} \leq \exp\left\{\frac{d_x}{4}\log\left(\frac{1}{m\sigma_\nu^2(2-M\sigma_\nu^2)}\right) + \frac{M}{2(2-M\sigma_\nu^2)}\|\bar{\mathrm{x}}-\mu_\nu\|_2^2\right\}. \end{aligned} \quad (2.111)$$

The result is complete by incorporating (2.111) into the result (2.107). □

Lemma 2.5. *For $d\nu(\mathrm{x}) \sim \delta(\mathrm{x} - \mathrm{x}_\nu)d\mathrm{x}$, the functional derivative*

$$\frac{d\nu}{d\mathcal{P}_\pi} = \frac{1}{\int_{\Omega(\mathrm{x}_\nu)} \pi(\mathrm{x}')d\mathrm{x}'} \tag{2.112}$$

is analogous to the Radon-Nikodym, where $d\mathcal{P}_\pi(\mathrm{x}) = \pi(\mathrm{x})d\mathrm{x}$ and the integral is taken over the region

$$\Omega(\mathrm{x}_\nu) = \left\{\mathrm{x}' \in \mathbb{R}^{d_x} : \|\mathrm{x}' - \bar{\mathrm{x}}\|_2 \geq \|\mathrm{x}_\nu - \bar{\mathrm{x}}\|_2\right\}.$$

Proof. The Dirac-delta measure is singular, i.e. not absolutely continuous with respect to the Lebesgue measure, and hence its Radon-Nikodym derivative may not be formally defined in general. However, by abusing notation one can obtain a functional expression for the enunciated derivative. Observe the definition of the Radon-Nikodym derivative as

$$\frac{d\mathcal{P}_\pi}{d\nu} = f \implies \mathcal{P}_\pi(\mathcal{D}) = \int_{\mathcal{D}} f\, d\nu, \tag{2.113}$$

for any measurable domain $\mathcal{D} \subseteq \mathbb{R}^{d_x}$. Given $\nu(\mathrm{x}_\nu) = \delta(\mathrm{x}_\nu - \mathrm{x}) \equiv \delta(\mathrm{x} - \mathrm{x}_\nu) = \nu(\mathrm{x})$ and setting $f(\mathrm{x}) \coloneqq \int_{\Omega(\mathrm{x})} \pi(\mathrm{x}')d\mathrm{x}'$, we write

$$\mathcal{P}_\pi(\mathrm{x}) = \int_{\mathbb{R}^{d_x}} f(\mathrm{x}_\nu)\, d\nu(\mathrm{x}_\nu) = \int_{\mathbb{R}^{d_x}} f(\mathrm{x}_\nu)\, \delta(\mathrm{x}_\nu - \mathrm{x})d\mathrm{x}_\nu = f(\mathrm{x}) = \int_{\Omega(\mathrm{x})} \pi(\mathrm{x}')d\mathrm{x}', \tag{2.114}$$

where $\frac{d\mathcal{P}_\pi}{d\mathrm{x}} = \pi(\mathrm{x})$ accordingly, and $f(\mathrm{x}_\nu) = \int_{\Omega(\mathrm{x}_\nu)} \pi(\mathrm{x}')d\mathrm{x}'$. Notice that $f(\mathrm{x})$ plays the role of a cumulative distribution function that is integrated over the "tail" of $\pi(\mathrm{x}')$, in the region defined by $\Omega(\mathrm{x})$. Thus, applying definition (2.113), we get

$$\begin{aligned} \frac{d\mathcal{P}_\pi}{d\nu} &\coloneqq f(\mathrm{x}_\nu) = \int_{\Omega(\mathrm{x}_\nu)} \pi(\mathrm{x}')d\mathrm{x}', \\ \therefore \frac{d\nu}{d\mathcal{P}_\pi} &= \frac{1}{\int_{\Omega(\mathrm{x}_\nu)} \pi(\mathrm{x}')d\mathrm{x}'}. \end{aligned} \tag{2.115}$$

□

Remark 2.6. Notice that if $\pi(\mathrm{x}') = (2\pi P^{-1})^{-\frac{d_x}{2}} e^{-\frac{P}{2}\|\mathrm{x}' - \bar{\mathrm{x}}\|_2^2}$, for some $P \in \mathbb{R}_+$, we can compute the resulting integral in *Lemma 2.5* using the spherical symmetry of the Gaussian function to

give

$$\begin{aligned}
\int_{\Omega(\mathrm{x}_\nu)} \pi(\mathrm{x}')d\mathrm{x}' &= (2\pi P^{-1})^{-\frac{d_x}{2}} \int_{\Omega(\mathrm{x}_\nu)} e^{-\frac{P}{2}\|\mathrm{x}'-\bar{\mathrm{x}}\|_2^2} d\mathrm{x}' \\
&= (2\pi)^{-\frac{d_x}{2}} \int_{\{\|u\|_2 \geq \sqrt{P}\|\mathrm{x}_\nu-\bar{\mathrm{x}}\|_2\}} e^{-\frac{1}{2}\|u\|_2^2} du \\
&= (2\pi)^{-\frac{d_x}{2}} \int_{r_\nu}^{\infty} \int_{\mathcal{S}^{d_x-1}(r)} e^{-\frac{1}{2}r^2} d\mathcal{A}\, dr \\
&= (2\pi)^{-\frac{d_x}{2}} \int_{r_\nu}^{\infty} e^{-\frac{1}{2}r^2} \mathcal{A}_{d_x-1}(r)\, dr \\
&= (2\pi)^{-\frac{d_x}{2}} \mathcal{A}_{d_x-1}(1) \int_{r_\nu}^{\infty} e^{-\frac{1}{2}r^2} r^{d_x-1}\, dr \\
&= (2\pi)^{-\frac{d_x}{2}} \frac{2\pi^{\frac{d_x}{2}}}{\Gamma\left(\frac{d_x}{2}\right)} \int_{r_\nu}^{\infty} e^{-\frac{1}{2}r^2} r^{d_x-1}\, dr \\
&= (2\pi)^{-\frac{d_x}{2}} \frac{2\pi^{\frac{d_x}{2}}}{\Gamma\left(\frac{d_x}{2}\right)} 2^{\frac{d_x}{2}-1} \int_{r_\nu^2/2}^{\infty} e^{-t} \cdot t^{\frac{d_x}{2}-1}\, dt \\
&= \frac{1}{\Gamma\left(\frac{d_x}{2}\right)} \int_{r_\nu^2/2}^{\infty} t^{\frac{d_x}{2}-1} e^{-t}\, dt \\
&= \Gamma_u\left(\frac{d_x}{2}, \frac{r_\nu^2}{2}\right) = \Gamma_u\left(\frac{d_x}{2}, \frac{P\|\mathrm{x}_\nu - \bar{\mathrm{x}}\|_2^2}{2}\right),
\end{aligned}$$

where $du = d\mathcal{A}\, dr$ for a volume element $du \in \mathbb{R}^{d_x}$, area element $d\mathcal{A} \in \mathbb{R}^{d_x-1}$ and radius element $dr \in \mathbb{R}_+$. In addition, $\mathcal{S}^{d_x-1}(r)$ denotes the (d_x-1)-sphere of radius r, with a total surface area of $\mathcal{A}_{d_x-1}(r)$, and the recursion $\mathcal{A}_{d_x-1}(r) = \mathcal{A}_{d_x-1}(1)r^{d_x-1}$ has been applied with $\mathcal{A}_{d_x-1}(1) = 2\pi^{\frac{d_x}{2}}/\Gamma(d_x/2)$. The lower integration extreme is taken as $r_\nu \coloneqq \sqrt{P}\|\mathrm{x}_\nu - \bar{\mathrm{x}}\|_2$, and $\Gamma_u(s,x) = \Gamma(s)^{-1}\int_x^\infty t^{s-1}e^{-t}dt$ is the upper incomplete gamma function. Also, we recall that $\int_{\Omega(\mathrm{x}_\nu)} \pi(\mathrm{x}')d\mathrm{x}' = 1 - \int_{\mathbb{R}^{d_x}\setminus\Omega(\mathrm{x}_\nu)} \pi(\mathrm{x}')d\mathrm{x}'$.

Lemma 2.7. *Under conditions (2.102), (2.103) and (2.104), and assumptions of Lemma 2.3, given a initial probability mass located at* $\mathrm{x} = \mathrm{x}_\nu$*, i.e.* $\nu(\mathrm{x}) = \delta(\mathrm{x} - \mathrm{x}_\nu)$ *(Dirac delta), then*

$$\|\mathcal{P}_{\mathcal{L},T} - \mathcal{P}_\pi\|_{TV} \leq \frac{1}{2}\exp\left\{-\frac{1}{2}mT + \frac{d_x}{2}\log\left(\frac{M}{m}\right) - \log\left[\Gamma_u\left(\frac{d_x}{2}, \frac{M\|\bar{\mathrm{x}} - \mathrm{x}_\nu\|_2^2}{2}\right)\right]\right\}. \quad (2.116)$$

Proof. Considering $\nu(\mathrm{x}) = \delta(\mathrm{x} - \mathrm{x}_\nu)$ on the definition of total variation distance, we use

Lemma 2.5 and *Remark 2.6.* The procedure follows as

$$
\begin{aligned}
\|\mathcal{P}_{\mathcal{L},T} - \mathcal{P}_\pi\|_{\mathrm{TV}} &= \sup_{A\in\mathscr{B}(\mathbb{R}^{d_x})} \left| \int_{\mathbb{R}^{d_x}} P_t(\mathrm{x}, A) d\nu(\mathrm{x}) - \mathcal{P}_\pi(A) \right| \\
&= \sup_{A\in\mathscr{B}(\mathbb{R}^{d_x})} \left| \int_{\mathbb{R}^{d_x}} \left(P_t(\mathrm{x}, A) - \mathcal{P}_\pi(A)\right) d\nu(\mathrm{x}) \right| \\
&= \sup_{A\in\mathscr{B}(\mathbb{R}^{d_x})} \left| \int_{\mathbb{R}^{d_x}} \left(P_t(\mathrm{x}, A) - \mathcal{P}_\pi(A)\right) \frac{d\nu}{d\mathcal{P}_\pi} d\mathcal{P}_\pi(\mathrm{x}) \right| \\
&= \sup_{A\in\mathscr{B}(\mathbb{R}^{d_x})} \left| \int_{\mathbb{R}^{d_x}} \left(P_t(\mathrm{x}, A) - \mathcal{P}_\pi(A)\right) \frac{1}{\int_{\Omega(\mathrm{x}_\nu)} \pi(\mathrm{x}')d\mathrm{x}'} d\mathcal{P}_\pi(\mathrm{x}) \right| \\
&\leq \frac{1}{\int_{\Omega(\mathrm{x}_\nu)} \pi(\mathrm{x}')d\mathrm{x}'} \cdot \sup_{A\in\mathscr{B}(\mathbb{R}^{d_x})} \int_{\mathbb{R}^{d_x}} \left|P_t(\mathrm{x}, \mathcal{A}) - \mathcal{P}_\pi(A)\right| \pi(\mathrm{x}) d\mathrm{x} \\
&\leq \frac{1}{\int_{\Omega(\mathrm{x}_\nu)} \pi(\mathrm{x}')d\mathrm{x}'} \cdot \sup_{A\in\mathscr{B}(\mathbb{R}^{d_x})} \left[\int_{\mathbb{R}^{d_x}} \left|P_t(\mathrm{x}, A) - \mathcal{P}_\pi(A)\right|^2 \pi(\mathrm{x}) d\mathrm{x} \right]^{\frac{1}{2}} \\
&\leq \frac{\left(2\pi m^{-1}\right)^{d_x/2}}{\int_{\Omega(\mathrm{x}_\nu)} e^{-\frac{M}{2}\|\mathrm{x}'-\bar{\mathrm{x}}\|_2^2} d\mathrm{x}'} \sup_{A\in\mathscr{B}(\mathbb{R}^{d_x})} \left[\int_{\mathbb{R}^{d_x}} \left|P_t(\mathrm{x}, A) - \mathcal{P}_\pi(A)\right|^2 \pi(\mathrm{x}) d\mathrm{x} \right]^{\frac{1}{2}} \\
&= \frac{\left(2\pi m^{-1}\right)^{d_x/2} \left(2\pi M^{-1}\right)^{-d_x/2}}{\int_{\Omega(\mathrm{x}_\nu)} \mathcal{N}(\mathrm{x}'; \bar{\mathrm{x}}, M^{-1}\mathbb{I}_{d_x}) d\mathrm{x}'} \sup_{A\in\mathscr{B}(\mathbb{R}^{d_x})} \left[\int_{\mathbb{R}^{d_x}} \left|P_t(\mathrm{x}, A) - \mathcal{P}_\pi(A)\right|^2 \pi(\mathrm{x}) d\mathrm{x} \right]^{\frac{1}{2}} \\
&= \frac{\left(Mm^{-1}\right)^{d_x/2}}{\Gamma_u\left(\frac{d_x}{2}, \frac{M\|\bar{\mathrm{x}}-\mathrm{x}_\nu\|_2^2}{2}\right)} \sup_{A\in\mathscr{B}(\mathbb{R}^{d_x})} \left[\int_{\mathbb{R}^{d_x}} \left|P_t(\mathrm{x}, A) - \mathcal{P}_\pi(A)\right|^2 \pi(\mathrm{x}) d\mathrm{x} \right]^{\frac{1}{2}} \\
\text{(Dalalyan, 2014)} &\leq \left(\frac{M}{m}\right)^{d_x/2} \cdot \Gamma_u\left(\frac{d_x}{2}, \frac{M\|\mathrm{x}_\nu - \bar{\mathrm{x}}\|_2^2}{2}\right)^{-1} \cdot \frac{1}{2} e^{-\frac{1}{2}mT} \\
&= \frac{1}{2} \exp\left\{ -\frac{1}{2}mT + \frac{d_x}{2} \log\left(\frac{M}{m}\right) - \log\left[\Gamma_u\left(\frac{d_x}{2}, \frac{M\|\bar{\mathrm{x}} - \mathrm{x}_\nu\|_2^2}{2}\right) \right] \right\}. \quad (2.117)
\end{aligned}
$$

□

Lemma 2.8. *Suppose a probability measure $\mathcal{P}_{\tilde{\mathcal{L}}[\Delta\lambda],T}$ produced by numerical integration with step size $\Delta\lambda$, up to time horizon $T = L\Delta\lambda$, of the Langevin diffusion SDE according to*

$$
\begin{aligned}
d\tilde{\mathcal{L}}_\lambda = -\frac{1}{2} \sum_{l=0}^{L-1} \mathrm{D}_{l\Delta\lambda} \nabla\tilde{\Phi}(\tilde{\mathcal{L}}_{l\Delta\lambda}) \mathbb{1}_{[l\Delta\lambda,(l+1)\Delta\lambda)} d\lambda \\
+ \sum_{l=0}^{L-1} \mathrm{D}_{l\Delta\lambda}^{1/2} d\mathrm{W}_\lambda,\ 0 \leq \lambda \leq L\Delta\lambda,\ \tilde{\mathcal{L}}_0 = 0. \quad (2.118)
\end{aligned}
$$

The resulting Markov chain is assumed to depart from the initial density $\nu(\mathrm{x}_0) = \mathcal{N}(\mathrm{x}_0; \mu_\nu, \sigma_\nu^2\mathbb{I}_{d_x})$ and targets the invariant density $\pi(\mathrm{x}) \propto \exp\{-\Phi(\mathrm{x})\}$ with measure $\mathcal{P}_\pi(d\mathrm{x})$. The process $\{\mathrm{W}_\lambda\}_{\lambda\geq 0}$ is the standard Wiener process and the diffusion matrix $\mathrm{D}_\lambda = \mathrm{D}(\mathrm{x}_\lambda)$ is bounded by $K = \sup_{\mathrm{x}} \|\mathrm{D}(\mathrm{x})\|_2$. Under assumptions (2.102), (2.103), and (2.104), then

$$
\begin{aligned}
\|\mathcal{P}_{\mathcal{L},T} - \mathcal{P}_{\tilde{\mathcal{L}}[\Delta\lambda],T}\|_{TV} \leq \frac{1}{2} \\
- \frac{1}{2} \exp\left\{ -\frac{d_x}{2} \frac{M^3K^4\gamma}{48(2\gamma-1)} \left(\sigma_\nu^2 + \frac{1}{d_x}\|\bar{\mathrm{x}} - \mu_\nu\|_2^2 + 2T\right) \Delta\lambda^2 - \frac{d_x M^2 K^3 T}{16} \Delta\lambda \right\}, \quad (2.119)
\end{aligned}
$$

for $T \geq 0$ and where $\mathcal{P}_{\mathcal{L},T}$ is the probability measure produced by the exact integration of the Langevin diffusion SDE.

Proof. We take a different approach as that proposed by Dalalyan [41] for this proof. Instead of bounding the total variance distance $\|\mathcal{P}_{\mathcal{L},T} - \mathcal{P}_{\tilde{\mathcal{L}}[\Delta\lambda],T}\|_{\text{TV}}$ by the Pinsker inequality, we treat it directly. First, identify that

$$\begin{aligned}
\|\mathcal{P}_{\mathcal{L},T} - \mathcal{P}_{\tilde{\mathcal{L}}[\Delta\lambda],T}\|_{\text{TV}} &= \frac{1}{2}\int_{\mathbb{R}^{d_x}} \left| d\mathcal{P}_{\mathcal{L},T} - d\mathcal{P}_{\tilde{\mathcal{L}}[\Delta\lambda],T} \right| \\
&= \frac{1}{2}\int_{\mathbb{R}^{d_x}} \left| 1 - \frac{d\mathcal{P}_{\tilde{\mathcal{L}}[\Delta\lambda],T}}{d\mathcal{P}_{\mathcal{L},T}} \right| d\mathcal{P}_{\mathcal{L},T} \\
&= \frac{1}{2}\int_{\mathbb{R}^{d_x}} \left| 1 - \frac{\nu(\mathrm{x}_0) dP_t^{\tilde{\mathcal{L}}}(\mathrm{x}_0, \mathrm{x})}{\nu(\mathrm{x}_0) dP_t^{\mathcal{L}}(\mathrm{x}_0, \mathrm{x})} \right| d\mathcal{P}_{\mathcal{L},T} \\
&= \frac{1}{2}\int_{\mathbb{R}^{d_x}} \left| 1 - \frac{dP_t^{\tilde{\mathcal{L}}}(\mathrm{x}_0, \mathrm{x})}{dP_t^{\mathcal{L}}(\mathrm{x}_0, \mathrm{x})} \right| d\mathcal{P}_{\mathcal{L},T}(\mathrm{x}) \\
&= \frac{1}{2}\int_{\mathbb{R}^{d_x}} \mathbb{E}^{P_t^{\mathcal{L}}} \left[\left| 1 - \frac{dP_t^{\tilde{\mathcal{L}}}(\mathrm{x}_0, \mathrm{x})}{dP_t^{\mathcal{L}}(\mathrm{x}_0, \mathrm{x})} \right| |\mathrm{x}_0 \right] d\nu(\mathrm{x}_0),
\end{aligned} \tag{2.120}$$

where $\mathcal{P}_{\mathcal{L},T}(d\mathrm{x}) = \int_{\mathbb{R}^{d_x}} P_t^{\mathcal{L}}(\mathrm{x}_0, d\mathrm{x})\nu(\mathrm{x}_0)d\mathrm{x}_0$, and $P_t^{\tilde{\mathcal{L}}}(\mathrm{x}_0, \cdot)$ and $P_t^{\mathcal{L}}(\mathrm{x}_0, \cdot)$ are the Markov transition kernels for the discrete-time and continuous-time Langevin dynamics respectively. By applying Girsanov's theorem to change the measure from $\mathcal{P}_{\mathcal{L},T}$ to $\mathcal{P}_{\tilde{\mathcal{L}}[\Delta\lambda],T}$, one obtains (see step 3 of the proof of *Proposition 2* in [42])

$$\begin{aligned}
&\frac{dP_t^{\tilde{\mathcal{L}}}}{dP_t^{\mathcal{L}}}(\mathrm{x}) \\
&= \exp\left\{ \int_0^T \tfrac{1}{2} \left(-\textstyle\sum_{l=0}^{L-1} \mathrm{D}_{l\Delta\lambda} \nabla\tilde{\Phi}(\mathrm{x}_{l\Delta\lambda}) \mathbb{1}_{[l\Delta\lambda,(l+1)\Delta\lambda)} + \mathrm{D}_\lambda \nabla\Phi(\mathrm{x}_\lambda) \right)^T \mathrm{D}_\lambda^{-1} \left(d\mathcal{L}_\lambda + \tfrac{1}{2}\mathrm{D}_\lambda \nabla\Phi(\mathcal{L}_\lambda) d\lambda \right) \right\} \\
&\quad \times \exp\left\{ -\frac{1}{2}\int_0^T \left\| -\frac{1}{2}\sum_{l=0}^{L-1} \mathrm{D}_{l\Delta\lambda} \nabla\tilde{\Phi}(\mathrm{x}_{l\Delta\lambda}) \mathbb{1}_{[l\Delta\lambda,(l+1)\Delta\lambda)} + \frac{1}{2}\mathrm{D}_\lambda \nabla\Phi(\mathrm{x}_\lambda) \right\|_2^2 d\lambda \right\}.
\end{aligned} \tag{2.121}$$

Thus, since $\mathbb{E}[e^z] \geq e^{\mathbb{E}[z]}$ by Jensen's inequality, $\mathrm{D}_\lambda = \mathrm{D}(\mathrm{x}_\lambda) \leq K$, and $\nabla\Phi(\mathrm{x})$ is Lipschitz-continuous with constant M (2.104), we obtain

$$\begin{aligned}
&\mathbb{E}^{P_t^{\mathcal{L}}} \left[\frac{dP_t^{\tilde{\mathcal{L}}}}{dP_t^{\mathcal{L}}}(\mathrm{x}) | \mathrm{x}_0 \right] \\
&\geq \exp\left\{ -\mathbb{E}^{P_t^{\mathcal{L}}} \left[\frac{1}{2}\sum_{l=0}^{L-1} \int_{l\Delta\lambda}^{(l+1)\Delta\lambda} \left\| \frac{1}{2}\mathrm{D}_\lambda \nabla\Phi(\mathrm{x}_\lambda) - \frac{1}{2}\mathrm{D}_{l\Delta\lambda} \nabla\tilde{\Phi}(\mathrm{x}_{l\Delta\lambda}) \right\|_2^2 d\lambda | \mathrm{x}_0 \right] \right\} \\
&= \exp\left\{ -\frac{1}{2}\sum_{l=0}^{L-1} \int_{l\Delta\lambda}^{(l+1)\Delta\lambda} \mathbb{E}^{P_t^{\mathcal{L}}} \left[\left\| \frac{1}{2}\mathrm{D}_\lambda \nabla\Phi(\mathrm{x}_\lambda) - \frac{1}{2}\mathrm{D}_{l\Delta\lambda} \nabla\tilde{\Phi}(\mathrm{x}_{l\Delta\lambda}) \right\|_2^2 | \mathrm{x}_0 \right] d\lambda \right\} \\
&\geq \exp\left\{ -\frac{K^2}{8}\sum_{l=0}^{L-1} \int_{l\Delta\lambda}^{(l+1)\Delta\lambda} \mathbb{E}^{P_t^{\mathcal{L}}} \left[\left\| \nabla\Phi(\mathrm{x}_\lambda) - \nabla\tilde{\Phi}(\mathrm{x}_{l\Delta\lambda}) \right\|_2^2 | \mathrm{x}_0 \right] d\lambda \right\} \\
&\geq \exp\left\{ -\frac{K^2 M^2}{8}\sum_{l=0}^{L-1} \int_{l\Delta\lambda}^{(l+1)\Delta\lambda} \mathbb{E}^{P_t^{\mathcal{L}}} \left[\|\mathrm{x}_\lambda - \mathrm{x}_{l\Delta\lambda}\|_2^2 | \mathrm{x}_0 \right] d\lambda \right\}.
\end{aligned} \tag{2.122}$$

For each step $\mathrm{x}_\lambda - \mathrm{x}_{l\Delta\lambda} = -\frac{1}{2}\mathrm{D}_{l\Delta\lambda}\nabla\Phi(\mathrm{x}_{l\Delta\lambda})(\lambda - l\Delta\lambda) + \mathrm{D}^{1/2}_{l\Delta\lambda}(\mathrm{W}_\lambda - \mathrm{W}_{l\Delta\lambda})$, hence

$$\begin{aligned}
&\mathbb{E}^{P^{\mathcal{L}}_t}\left[\frac{dP^{\tilde{\mathcal{L}}}_t}{dP^{\mathcal{L}}_t}(\mathrm{x})|\mathrm{x}_0\right] \\
&\geq \exp\left\{-\frac{K^2M^2}{8}\sum_{l=0}^{L-1}\int_{l\Delta\lambda}^{(l+1)\Delta\lambda}\mathbb{E}^{P^{\mathcal{L}}_t}\left[\left\|\frac{1}{2}\mathrm{D}(\mathrm{x}_{l\Delta\lambda})\nabla\Phi(\mathrm{x}_{l\Delta\lambda})\right\|_2^2(\lambda - l\Delta\lambda)^2|\mathrm{x}_0\right]d\lambda\right\} \\
&\times \exp\left\{-\frac{K^2M^2}{8}\sum_{l=0}^{L-1}\int_{l\Delta\lambda}^{(l+1)\Delta\lambda}\mathbb{E}^{P^{\mathcal{L}}_t}\left[\left\|\mathrm{D}(\mathrm{x}_{l\Delta\lambda})^{1/2}(\mathrm{W}_\lambda - \mathrm{W}_{l\Delta\lambda})\right\|_2^2|\mathrm{x}_0\right]d\lambda\right\} \\
&\geq \exp\left\{-\frac{K^2M^2}{8}\sum_{l=0}^{L-1}\int_{l\Delta\lambda}^{(l+1)\Delta\lambda}\frac{K^2}{4}\mathbb{E}^{P^{\mathcal{L}}_t}\left[\|\nabla\Phi(\mathrm{x}_{l\Delta\lambda})\|_2^2|\mathrm{x}_0\right](\lambda - l\Delta\lambda)^2d\lambda\right\} \\
&\times \exp\left\{-\frac{K^2M^2}{8}\sum_{l=0}^{L-1}\int_{l\Delta\lambda}^{(l+1)\Delta\lambda}K(\lambda - l\Delta\lambda)d_xd\lambda\right\} \\
&= \exp\left\{-\frac{M^2K^4\Delta\lambda^3}{96}\sum_{l=0}^{L-1}\mathbb{E}^{P^{\mathcal{L}}_t}\left[\|\nabla\Phi(\mathrm{x}_{l\Delta\lambda})\|_2^2|\mathrm{x}_0\right] - \frac{d_xM^2K^3L\Delta\lambda^2}{16}\right\}. \qquad (2.123)
\end{aligned}$$

Now we invoke a result from *Corollary 4* in [41], that gives

$$\Delta\lambda\sum_{l=0}^{L-1}\mathbb{E}\left[\|\nabla\Phi(\mathrm{x}_{l\Delta\lambda})\|_2^2\right] \leq \frac{M\gamma}{2\gamma-1}\mathbb{E}\left[\|\mathrm{x}_0 - \bar{\mathrm{x}}\|_2^2\right] + \frac{2\gamma MTd_x}{2\gamma-1}, \qquad (2.124)$$

for some $\gamma > 1$, $\Delta\lambda \leq (\gamma M)^{-1}$ and $L > 1$. Substituting $L\Delta\lambda = T$ and incorporating (2.124):

$$\begin{aligned}
&\mathbb{E}^{P^{\mathcal{L}}_t}\left[\frac{dP^{\tilde{\mathcal{L}}}_t}{dP^{\mathcal{L}}_t}(\mathrm{x})|\mathrm{x}_0\right] \\
&\geq \exp\left\{-\frac{M^2K^4\Delta\lambda^2}{48}\frac{\gamma}{2\gamma-1}\left(\frac{M}{2}\mathbb{E}^{P^{\mathcal{L}}_t}\left[\|\mathrm{x}_0 - \bar{\mathrm{x}}\|_2^2|\mathrm{x}_0\right] + d_xMT\right) - \frac{d_xM^2K^3T\Delta\lambda}{16}\right\} \\
&= \exp\left\{-\frac{M^2K^4\Delta\lambda^2}{48}\frac{\gamma}{2\gamma-1}\left(\frac{M}{2}\|\mathrm{x}_0 - \bar{\mathrm{x}}\|_2^2 + d_xMT\right) - \frac{d_xM^2K^3T\Delta\lambda}{16}\right\}. \qquad (2.125)
\end{aligned}$$

Since

$$0 \leq \mathbb{E}^{P^{\mathcal{L}}_t}\left[\frac{dP^{\tilde{\mathcal{L}}}_t}{dP^{\mathcal{L}}_t}(\mathrm{x})|\mathrm{x}_0\right] \leq 1,$$

and applying expression (2.125) in (2.120), we have

$$\begin{aligned}
&\|\mathcal{P}_{\mathcal{L},T} - \mathcal{P}_{\tilde{\mathcal{L}}[\Delta\lambda],T}\|_{\mathrm{TV}} \\
&= \frac{1}{2}\int_{\mathbb{R}^{d_x}}\left|1 - \frac{dP^{\tilde{\mathcal{L}}}_t(\mathrm{x}_0,\mathrm{x})}{dP^{\mathcal{L}}_t(\mathrm{x}_0,\mathrm{x})}\right|d\mathcal{P}_{\mathcal{L},T}(\mathrm{x}) \\
&= \frac{1}{2}\int_{\mathbb{R}^{d_x}}d\mathcal{P}_{\mathcal{L},T}(\mathrm{x}) - \frac{1}{2}\int_{\mathbb{R}^{d_x}}\mathbb{E}^{P^{\mathcal{L}}_t}\left[\frac{dP^{\tilde{\mathcal{L}}}_t(\mathrm{x}_0,\mathrm{x})}{dP^{\mathcal{L}}_t(\mathrm{x}_0,\mathrm{x})}|\mathrm{x}_0\right]d\nu(\mathrm{x}_0) \\
&\leq \frac{1}{2} - \frac{1}{2}\int_{\mathbb{R}^{d_x}}e^{-\frac{M^2K^4\Delta\lambda^2}{48}\frac{\gamma}{2\gamma-1}\left(\frac{M}{2}\|\mathrm{x}_0-\bar{\mathrm{x}}\|_2^2 + d_xMT\right) - \frac{d_xM^2K^3T\Delta\lambda}{16}}\nu(\mathrm{x}_0)d\mathrm{x}_0. \qquad (2.126)
\end{aligned}$$

Finally, using the substitutions $A_1 = \frac{M^3K^4}{96}\frac{\gamma}{2\gamma-1}$, $A_2 = \frac{d_xM^3TK^4}{48}\frac{\gamma}{2\gamma-1}$, $B = \frac{d_xM^2K^3T}{16}$,

$\mathbb{E}\left[e^{-z}\right] \geq e^{-\mathbb{E}[z]}$ by Jensen's inequality, and $\nu(\mathrm{x}_0) = \mathcal{N}(\mathrm{x}_0;\, \mu_\nu, \sigma_\nu^2 \mathbb{I}_{d_x})$, we get

$$\begin{aligned}
\|\mathcal{P}_{\mathcal{L},T} - \mathcal{P}_{\tilde{\mathcal{L}}[\Delta\lambda],T}\|_{\mathrm{TV}} &\leq \frac{1}{2} - \frac{1}{2} e^{-A_2\Delta\lambda^2 - B\Delta\lambda} \int_{\mathbb{R}^{d_x}} e^{-A_1\Delta\lambda^2 \|\mathrm{x}_0 - \bar{\mathrm{x}}\|_2^2} \nu(\mathrm{x}_0) d\mathrm{x}_0 \\
&\leq \frac{1}{2} - \frac{1}{2} e^{-A_2\Delta\lambda^2 - B\Delta\lambda} e^{-A_1\Delta\lambda^2 \int_{\mathbb{R}^{d_x}} \|\mathrm{x}_0 - \bar{\mathrm{x}}\|_2^2 \nu(\mathrm{x}_0) d\mathrm{x}_0} \\
&= \frac{1}{2} - \frac{1}{2} e^{-A_2\Delta\lambda^2 - B\Delta\lambda - A_1\Delta\lambda^2 \left(d_x \sigma_\nu^2 + \|\bar{\mathrm{x}} - \mu_\nu\|_2^2\right)} \\
&= \frac{1}{2} - \frac{1}{2} e^{-\left(A_1\left(d_x\sigma_\nu^2 + \|\bar{\mathrm{x}} - \mu_\nu\|_2^2\right) + A_2\right)\Delta\lambda^2 - B\Delta\lambda} \\
&= \frac{1}{2} - \frac{1}{2} e^{-\frac{d_x}{2}\frac{M^3K^4\gamma}{48(2\gamma-1)}\left(\sigma_\nu^2 + \frac{1}{d_x}\|\bar{\mathrm{x}} - \mu_\nu\|_2^2 + 2T\right)\Delta\lambda^2 - \frac{d_x M^2 K^3 T}{16}\Delta\lambda}. \qquad (2.127)
\end{aligned}$$

□

Lemma 2.9. *Under the same assumptions as of Lemma 2.8, except for a initial density* $\nu(\mathrm{x}_0) = \delta(\mathrm{x}_0 - \mathrm{x}_\nu)$ *(Dirac delta), i.e. a probability mass initially located at* $\mathrm{x}_0 = \mathrm{x}_\nu$*, then*

$$\begin{aligned}
\|\mathcal{P}_{\mathcal{L},T} - \mathcal{P}_{\tilde{\mathcal{L}}[\Delta\lambda],T}\|_{TV} &\leq \frac{1}{2} \\
&- \frac{1}{2}\exp\left\{-\frac{d_x}{2}\frac{M^3K^4\gamma}{48(2\gamma-1)}\left(\frac{1}{d_x}\|\bar{\mathrm{x}} - \mathrm{x}_\nu\|_2^2 + 2T\right)\Delta\lambda^2 - \frac{d_x M^2 K^3 T}{16}\Delta\lambda\right\},\ T \geq 0; \qquad (2.128)
\end{aligned}$$

Proof. The proof follows straightforwardly by substituting $\nu(\mathrm{x}_0) = \delta(\mathrm{x}_0 - \mathrm{x}_\nu)$ into (2.126). □

Theorem 2.10. *Let a convex function* Φ *satisfy the general assumptions (2.102), (2.103) and (2.104). Suppose a discrete-time Langevin Monte Carlo algorithm integrates (2.25), targeting the invariant density* $\pi(\mathrm{x}) \propto \exp\{-\Phi(\mathrm{x})\}$ *with measure* $\mathcal{P}_\pi(d\mathrm{x})$*. In addition, assume that for some* $\gamma \geq 1$ *we have* $\Delta\lambda \leq (\gamma M)^{-1}$*, and* $K = \sup_{\mathrm{x}} \|\mathrm{D}(\mathrm{x})\|_2$ *where* $\mathrm{D}_\lambda = \mathrm{D}(\mathrm{x}_\lambda)$ *is the diffusion matrix. Then, for a time horizon,* T*, and step size,* $\Delta\lambda$*, the total-variation distance between the target measure* $\mathcal{P}_\pi$ *and the approximated measure* $\mathcal{P}_{\tilde{\mathcal{L}}(\Delta\lambda),T}$ *furnished by a discrete-time Langevin Monte Carlo algorithm with initial density* $\nu\,(\mathrm{x}) = \mathcal{N}(\mathrm{x}; \mu_\nu, \sigma_\nu^2 \mathbb{I}_{d_x})$*, for* $\sigma_\nu^2 < 2M^{-1}$*, satisfies*

$$\begin{aligned}
\|\mathcal{P}_{\tilde{\mathcal{L}}[\Delta\lambda],T} - \mathcal{P}_\pi\|_{TV} &\leq \frac{1}{2}\exp\left\{-\frac{1}{2}mT + \frac{d_x}{4}\log\left(\frac{1}{m\sigma_\nu^2(2 - M\sigma_\nu^2)}\right) + \frac{M}{2(2 - M\sigma_\nu^2)}\|\bar{\mathrm{x}} - \mu_\nu\|_2^2\right\} \\
&+ \frac{1}{2} - \frac{1}{2}\exp\left\{-\frac{d_x}{2}\frac{M^3K^4\gamma}{48(2\gamma-1)}\left(\sigma_\nu^2 + \frac{1}{d_x}\|\bar{\mathrm{x}} - \mu_\nu\|_2^2 + 2T\right)\Delta\lambda^2 - \frac{d_x M^2 K^3 T}{16}\Delta\lambda\right\}. \qquad (2.129)
\end{aligned}$$

Proof. The proof follows from the triangle inequality

$$\|\mathcal{P}_{\tilde{\mathcal{L}}[\Delta\lambda],T} - \mathcal{P}_\pi\|_{\mathrm{TV}} \leq \|\mathcal{P}_{\mathcal{L},T} - \mathcal{P}_\pi\|_{\mathrm{TV}} + \|\mathcal{P}_{\mathcal{L},T} - \mathcal{P}_{\tilde{\mathcal{L}}[\Delta\lambda],T}\|_{\mathrm{TV}}, \qquad (2.130)$$

on which we substitute the results of *Lemmas 2.4* and *2.8* to give the final result (2.129). □

Corollary 2.11. *Let $d_x \geq 2$, Φ satisfy (2.102), (2.103) and (2.104), and $\varepsilon \in (0, 1/2)$ be a desired precision level. Let the time horizon, T, and the step size, $\Delta\lambda$, be defined by*

$$T \geq \frac{2\log(1/\varepsilon) + \frac{d_x}{2}\log\left(\frac{1}{m\sigma_\nu^2(2-M\sigma_\nu^2)}\right) + \frac{M}{(2-M\sigma_\nu^2)}\|\bar{\mathrm{x}} - \mu_\nu\|_2^2}{m}, \tag{2.131}$$

$$\Delta\lambda \leq \frac{-\frac{T}{16} + \sqrt{\left(\frac{T}{16}\right)^2 + \frac{\gamma}{48(2\gamma-1)}\left(\sigma_\nu^2 + \frac{1}{d_x}\|\bar{\mathrm{x}} - \mu_\nu\|_2^2 + 2T\right)M^{-1}K^{-2}\left[\frac{2}{d_x}\log\left(\frac{1}{1-\varepsilon}\right)\right]}}{\frac{\gamma}{48(2\gamma-1)}\left(\sigma_\nu^2 + \frac{1}{d_x}\|\bar{\mathrm{x}} - \mu_\nu\|_2^2 + 2T\right)MK}, \tag{2.132}$$

where $\gamma \geq 1$. Then the resulting probability distribution of a Langevin Monte Carlo algorithm that integrates (2.25) after $L = \lceil T/\Delta\lambda \rceil$ steps, satisfies $\|\mathcal{P}_{\tilde{\mathcal{L}}[\Delta\lambda],T} - \mathcal{P}_\pi\|_{TV} \leq \varepsilon$.

Proof. Bound each term on the right-hand side of (2.130) by half of the required precision ε, i.e.

$$\|\mathcal{P}_{\mathcal{L},T} - \mathcal{P}_\pi\|_{\mathrm{TV}} \leq \varepsilon/2,\ \|\mathcal{P}_{\mathcal{L},T} - \mathcal{P}_{\tilde{\mathcal{L}}[\Delta\lambda],T}\|_{\mathrm{TV}} \leq \varepsilon/2.$$

For the first term:

$$\begin{aligned}
\frac{1}{2}\exp\left\{-\frac{1}{2}mT + \frac{d_x}{4}\log\left(\frac{1}{m\sigma_\nu^2(2-M\sigma_\nu^2)}\right) + \frac{M}{2(2-M\sigma_\nu^2)}\|\bar{\mathrm{x}} - \mu_\nu\|_2^2\right\} &\leq \frac{\varepsilon}{2},\\
-\frac{1}{2}mT + \frac{d_x}{4}\log\left(\frac{1}{m\sigma_\nu^2(2-M\sigma_\nu^2)}\right) + \frac{M}{2(2-M\sigma_\nu^2)}\|\bar{\mathrm{x}} - \mu_\nu\|_2^2 &\leq \log\varepsilon,\\
\frac{1}{2}mT \geq -\log\varepsilon + \frac{d_x}{4}\log\left(\frac{1}{m\sigma_\nu^2(2-M\sigma_\nu^2)}\right) + \frac{M}{2(2-M\sigma_\nu^2)}\|\bar{\mathrm{x}} - \mu_\nu\|_2^2,&\\
T \geq \frac{2\log(1/\varepsilon) + \frac{d_x}{2}\log\left(\frac{1}{m\sigma_\nu^2(2-M\sigma_\nu^2)}\right) + \frac{M}{(2-M\sigma_\nu^2)}\|\bar{\mathrm{x}} - \mu_\nu\|_2^2}{m}.&
\end{aligned} \tag{2.133}$$

And for the second term:

$$\begin{aligned}
\frac{1}{2} - \frac{1}{2}\exp\left\{-\overbrace{\frac{d_x}{2}\frac{M^3K^4\gamma}{48(2\gamma-1)}\left(\sigma_\nu^2 + \frac{1}{d_x}\|\bar{\mathrm{x}} - \mu_\nu\|_2^2 + 2T\right)}^{a\,(>0)}\Delta\lambda^2 - \overbrace{\frac{d_xM^2K^3T}{16}}^{b\,(>0)}\Delta\lambda\right\} &\leq \frac{\varepsilon}{2},\\
\exp\left\{-a\Delta\lambda^2 - b\Delta\lambda\right\} &\geq 1-\varepsilon,\\
-a\Delta\lambda^2 - b\Delta\lambda &\geq \log(1-\varepsilon),\\
a\Delta\lambda^2 + b\Delta\lambda &\leq -\log(1-\varepsilon),\\
a\Delta\lambda^2 + b\Delta\lambda + c \leq 0,\ c = \log(1-\varepsilon) &= -\log\left(\frac{1}{1-\varepsilon}\right),\\
0 < \Delta\lambda \leq \frac{-b + \sqrt{b^2 - 4ac}}{2a},&\\
\Delta\lambda \leq \frac{-\frac{d_xM^2K^3T}{16} + \sqrt{\left(\frac{d_xM^2K^3T}{16}\right)^2 + 4\frac{d_x}{2}\frac{M^3K^4\gamma}{48(2\gamma-1)}\left(\sigma_\nu^2 + \frac{1}{d_x}\|\bar{\mathrm{x}} - \mu_\nu\|_2^2 + 2T\right)\log\left(\frac{1}{1-\varepsilon}\right)}}{2\frac{d_x}{2}\frac{M^3K^4\gamma}{48(2\gamma-1)}\left(\sigma_\nu^2 + \frac{1}{d_x}\|\bar{\mathrm{x}} - \mu_\nu\|_2^2 + 2T\right)},&\\
\Delta\lambda \leq \frac{-\frac{T}{16} + \sqrt{\left(\frac{T}{16}\right)^2 + \frac{\gamma}{48(2\gamma-1)}\left(\sigma_\nu^2 + \frac{1}{d_x}\|\bar{\mathrm{x}} - \mu_\nu\|_2^2 + 2T\right)M^{-1}K^{-2}\left[\frac{2}{d_x}\log\left(\frac{1}{1-\varepsilon}\right)\right]}}{\frac{\gamma}{48(2\gamma-1)}\left(\sigma_\nu^2 + \frac{1}{d_x}\|\bar{\mathrm{x}} - \mu_\nu\|_2^2 + 2T\right)MK}.&
\end{aligned} \tag{2.134}$$

□

Proof of Theorem 2.1

Proof of Theorem 2.1. We follow the same procedure established for proving *Theorem 2.10*, substituting the results of *Lemmas 2.7* and *2.9* into (2.130) to give

$$\|\mathcal{P}_{\tilde{\mathcal{L}}[\Delta\lambda],T} - \mathcal{P}_{\pi}\|_{\mathrm{TV}} \leq \frac{1}{2}\exp\left\{-\frac{1}{2}mT + \frac{d_x}{2}\log\left(\frac{M}{m}\right) - \log\left[\Gamma_u\left(\frac{d_x}{2}, \frac{M\|\bar{\mathrm{x}} - \mathrm{x}_\nu\|_2^2}{2}\right)\right]\right\}$$
$$+ \frac{1}{2} - \frac{1}{2}\exp\left\{-\frac{d_x}{2}\frac{M^3K^4\gamma}{48(2\gamma-1)}\left(\frac{1}{d_x}\|\bar{\mathrm{x}} - \mathrm{x}_\nu\|_2^2 + 2T\right)\Delta\lambda^2 - \frac{d_x M^2 K^3 T}{16}\Delta\lambda\right\}. \quad (2.135)$$

□

Proof of Corollary 2.2

Proof of Corollary 2.2. We follow the same procedure established for proving *Corollary 2.11*, bounding each term on the right-hand side of (2.130) in view of (2.135) to obtain

$$T \geq \frac{2\log(1/\varepsilon) + d_x\log\left(\frac{M}{m}\right) - 2\log\left[\Gamma_u\left(\frac{d_x}{2}, \frac{M\|\bar{\mathrm{x}} - \mathrm{x}_\nu\|_2^2}{2}\right)\right]}{m}, \quad (2.136)$$

$$\Delta\lambda \leq \frac{-\frac{T}{16} + \sqrt{\left(\frac{T}{16}\right)^2 + \frac{\gamma}{48(2\gamma-1)}\left(\frac{1}{d_x}\|\bar{\mathrm{x}} - \mathrm{x}_\nu\|_2^2 + 2T\right)M^{-1}K^{-2}\left[\frac{2}{d_x}\log\left(\frac{1}{1-\varepsilon}\right)\right]}}{\frac{\gamma}{48(2\gamma-1)}\left(\frac{1}{d_x}\|\bar{\mathrm{x}} - \mathrm{x}_\nu\|_2^2 + 2T\right)MK}. \quad (2.137)$$

□

2.9.2 On the Filtering Properties of the Stochastic Particle Flow

Theorem 2.12. *Define* $\mathrm{x} \in \mathbb{R}^{d_x}$ *to describe an* d_x*-dimensional vector state. Let the vector field* $\mu : \mathbb{R}^{d_x} \to \mathbb{R}^{d_x}$, $\mu(\mathrm{x}) \in \mathcal{C}^1(\mathbb{R}^{d_x})$*, be a conservative field, i.e. there exists a scalar potential function* $\psi : \mathbb{R}^{d_x} \to \mathbb{R}$, $\psi(\mathrm{x}) \in \mathcal{C}^2(\mathbb{R}^{d_x})$*, such that*

$$\mu(\mathrm{x}) = -\nabla_{\mathrm{x}}\psi(\mathrm{x}). \quad (2.138)$$

Let $p(\mathrm{x}, \lambda)$ *be the density of an ensemble of particles and, without loss of generality, can be assumed to be a continuous probability density function on* $\mathbb{R}^{d_x}$ *that depends on the pseudo-time variable* $\lambda \in \mathbb{R}$, $\lambda \geq 0$*. Set* $\pi(\mathrm{x}) \propto e^{-\psi(\mathrm{x})}$ *to be an invariant, locally log-concave probability density to which the density* $p(\mathrm{x}, \lambda)$ *is expected to converge weakly at a stationary state in a finite time horizon* $\lambda \geq T$, $T \in \mathbb{R}_+$*, i.e.*

$$\mathbb{E}_p[\varphi(\mathrm{x})] \to \mathbb{E}_\pi[\varphi(\mathrm{x})], \quad \lambda \to T; \quad (2.139)$$

for all bounded, continuous functions φ*, and where* $\mathbb{E}_p[.]$ *is the expectation with respect to the probability density* $p(\mathrm{x}, \lambda)$*. If the probability density* $p(\mathrm{x}, \lambda)$ *satisfies the continuity equation (Liouville's equation)*

$$\frac{\partial p}{\partial \lambda} = -\nabla_{\mathrm{x}} \cdot (p\,\mu), \quad \lambda \geq 0; \quad (2.140)$$

with the initial condition

$$p(\mathrm{x}, \lambda) = p_0(\mathrm{x}), \quad \lambda = 0; \quad (2.141)$$

then any probability mass (particle) $\mathrm{x}_m(0) \sim p_0(\mathrm{x})$*, when evolved according to the associated state equation*

$$d\mathrm{x}_m(\lambda) = \mu(\mathrm{x}_m(\lambda))\, d\lambda, \quad \lambda \geq 0; \tag{2.142}$$

converges to

$$\mathrm{x}_m(T) = argmax[\pi(\mathrm{x})], \quad \lambda \geq T, \tag{2.143}$$

at a stable equilibrium.

Proof. The general solution of the continuity equation without sources (2.140) assumes the form (see for example [116])

$$\begin{aligned} p(\mathrm{x}, \lambda) &= p_0(\mathrm{x}_m(\mathrm{x}, \lambda)) \left| \frac{\partial \mathrm{x}_m}{\partial \mathrm{x}} \right| \\ &= p_0(\mathrm{x}_m(\mathrm{x}, \lambda)) \left| \mathcal{J}_{\mathrm{x}}[\mathrm{x}_m(\mathrm{x}, \lambda)] \right|, \end{aligned} \tag{2.144}$$

where $\mathrm{x}_m(\mathrm{x}, \lambda)$ is an arbitrary element of mass that is regarded as a function of the pseudo-time λ and of the state x that it can possibly reach. The matrix $\mathcal{J}_{\mathrm{x}}[\mathrm{x}_m(\mathrm{x}, \lambda)]$ is the Jacobian matrix of $\mathrm{x}_m(\mathrm{x}, \lambda)$ with respect to x. Conceptually, at the stationary state $\mathrm{x}_m(\mathrm{x}_T, T) = \mathrm{x}_T$ the continuity equation (2.140) reads

$$\frac{\partial p}{\partial \lambda} = 0, \quad \lambda \geq T. \tag{2.145}$$

Using the general solution (2.144) to verify the stationary condition (2.145), we conclude that

$$p_0(\mathrm{x}_m(\mathrm{x}_T, T)) \left| \mathcal{J}_{\mathrm{x}}[\mathrm{x}_m(\mathrm{x}_T, T)] \right|$$

must be constant with respect to the pseudo-time, thus

$$\frac{d\mathrm{x}_m(\lambda)}{d\lambda} = \mu(\mathrm{x}_m(\lambda)) = 0, \quad \lambda \geq T. \tag{2.146}$$

Following the assumption of conservative field, $\mu(\mathrm{x}_T) = -\nabla_{\mathrm{x}}\psi(\mathrm{x}_T) = 0$ implies that the stationary state x_T is an equilibrium point, i.e. an extreme of the potential function ψ. In addition, since the potential function is assumed to be related to the stationary distribution as $\psi(\mathrm{x}) \propto -\log \pi(\mathrm{x})$, the stationary state x_T is an extreme of the stationary density.

A valid Lyapunov function of the flow is $V(\mathrm{x}) = \psi(\mathrm{x})$, which is positive semi-definite ($\psi(\mathrm{x}) \geq 0$) in the neighborhood of the equilibrium point due to the local log-concavity of the invariant density $\pi(\mathrm{x})$. Analyzing the (Lie) time derivative of the Lyapunov function in the neighborhood of the equilibrium point, $\|\mathrm{x} - \mathrm{x}_T\| < \varepsilon$ for a sufficiently small $\varepsilon \in \mathbb{R}_+$, we have

$$\begin{aligned} \frac{dV(\mathrm{x})}{d\lambda} &= \nabla_{\mathrm{x}} V(\mathrm{x})^T \cdot \frac{d\mathrm{x}}{d\lambda} = \nabla_{\mathrm{x}} V(\mathrm{x})^T \cdot \mu(\mathrm{x}), \\ \dot{V}(\mathrm{x}) &= \nabla_{\mathrm{x}} \psi(\mathrm{x})^T \cdot (-\nabla_{\mathrm{x}} \psi(\mathrm{x})), \\ \dot{V}(\mathrm{x}) &= -\|\nabla_{\mathrm{x}} \psi(\mathrm{x})\|^2 \leq 0, \ \|\mathrm{x} - \mathrm{x}_T\| < \varepsilon; \end{aligned} \tag{2.147}$$

from which we conclude that x_T is a point of (uniformly) stable equilibrium. Therefore, under the established hypotheses, any arbitrary probability mass $\mathrm{x}_m(\lambda)$ evolved according to (2.142)

converges to

$$\begin{aligned}\mathrm{x}_m(T) &= \operatorname{argmin}\left[\psi(\mathrm{x})\right] = \operatorname{argmin}\left[-\log\pi(\mathrm{x})\right],\\ \mathrm{x}_m(T) &= \operatorname{argmax}\left[\pi(\mathrm{x})\right], \quad \lambda \geq T;\end{aligned}$$

at a stable equilibrium. □

Lemma 2.13. *Let* $\{\mathrm{X}_\lambda : t \leq \lambda \leq T\}$ *be a diffusion process in* $\mathbb{R}^{d_x}$ *(hence a Markov process), solution of*

$$\begin{aligned} d\mathrm{X}_\lambda &= \mu_f(\mathrm{X}_\lambda, \lambda)\, d\lambda + \mathrm{D}_f(\lambda)^{1/2}\, d\mathrm{W}_\lambda, \\ \mathrm{X}_{\lambda=t} &= \mathrm{x}_t, t \leq \lambda \leq T; \end{aligned} \tag{2.148}$$

where $\{\mathrm{W}_\lambda : t \leq \lambda \leq T\}$ *is a standard Wiener process in* $\mathbb{R}^{d_x}$ *under the probability measure* $\mathcal{P}$*,* $\mu_f : \mathbb{R}^{d_x} \times [t,T] \to \mathbb{R}^{d_x}$ *is the drift and* $\mathrm{D}_f : [t,T] \to \mathbb{R}^{d_x \times d_x}$ *is a diffusion coefficient invariant over the space at any time instant. There exists an equivalent process* $\left\{\bar{\mathrm{X}}_\tau, \mathcal{V}_\tau : t \leq \tau \leq T\right\}$*, which is probabilistically the same as the original process, called* reverse process *on the interval* $[t,T]$ *(see [140]), that provides the solution of the stochastic system*

$$d\bar{\mathrm{X}}_\tau = \mu_r\left(\bar{\mathrm{X}}_\tau, \tau\right) d\tau + \mathrm{D}_r(\tau)^{1/2}\, d\bar{\mathrm{W}}_\tau,\ \bar{\mathrm{X}}_\tau(t) = \bar{\mathrm{x}}_t; \tag{2.149}$$

$$d\mathcal{V}_\tau = v_r\left(\bar{\mathrm{X}}_\tau, \tau\right)\mathcal{V}_\tau d\tau, \qquad \mathcal{V}_\tau(t) = 1; \tag{2.150}$$

for a standard Wiener process $\left\{\bar{\mathrm{W}}_\tau : t \leq \tau \leq T\right\}$ *in* $\mathbb{R}^{d_x}$ *under the measure* $\mathcal{P}$*, with the reverse drift and diffusion coefficients given, respectively, by*

$$\mu_r\left(\bar{\mathrm{X}}_\tau, \tau\right) = -\mu_f\left(\bar{\mathrm{X}}_\tau, T+t-\lambda\right), \tag{2.151}$$

$$\mathrm{D}_r(\tau) = \mathrm{D}_f(T+t-\lambda). \tag{2.152}$$

Proof. The Markov process $\{\mathrm{X}_\lambda\}_{\lambda\in[t,T]}$, as an existing solution to the SDE (2.148), has an associated probability density $p(\mathrm{x}_\lambda, \lambda)$ that must satisfy the Kolmogorov forward equation (Fökker-Planck equation):

$$\begin{aligned} \frac{\partial}{\partial\lambda}p &= -\nabla_{\mathrm{x}}\cdot(\mu_f p) + \frac{1}{2}\nabla_{\mathrm{x}}\cdot(\mathrm{D}_f\nabla_{\mathrm{x}}p),\ \lambda \geq t, \\ p(\mathrm{x}_{\lambda=t}, t) &= p_t(\mathrm{x}_t), \mathrm{x}_\lambda \in \mathbb{R}^{d_x}. \end{aligned}$$

The Fökker-Planck equation can be written in the non-divergence form as

$$\frac{\partial}{\partial\lambda}p = \hat{\mu}^T\nabla_{\mathrm{x}}p + \frac{1}{2}\nabla_{\mathrm{x}}\cdot(\mathrm{D}_f\nabla_{\mathrm{x}}p) + \hat{v}\cdot p, \tag{2.153}$$

where

$$\begin{aligned}\hat{\mu}(\mathrm{x}_\lambda) &= -\mu_f(\mathrm{x}_\lambda),\\ \hat{v}(\mathrm{x}_\lambda) &= -\nabla_{\mathrm{x}}\cdot\mu_f(\mathrm{x}_\lambda).\end{aligned}$$

We introduce the reverse time variable $\tau = T + t - \lambda$, so that

$$p\left(\mathrm{x}_{T+t-\lambda}, T+t-\lambda\right) \equiv \hat{p}\left(\bar{\mathrm{x}}_{\tau}, \tau\right),$$

and hence $-\partial_{\lambda} p = \partial_{\tau} \hat{p}$. Thus, rewriting (2.153) with respect to $\hat{p}\left(\bar{\mathrm{x}}_{\tau}, \tau\right)$ for $\tau \leq T$, $\bar{\mathrm{x}}_{\tau} \in \mathbb{R}^{d_x}$, and performing the substitutions

$$\begin{aligned} \mu_r\left(\bar{\mathrm{x}}_{\tau}, \tau\right) &= -\mu_f\left(\mathrm{x}_{T+t-\lambda}, T+t-\lambda\right), \\ \mathrm{D}_r\left(\tau\right) &= \quad \mathrm{D}_f\left(T+t-\lambda\right), \end{aligned}$$

we obtain

$$\begin{aligned} \frac{\partial}{\partial \lambda} p &= -\mu_f^T \nabla_{\mathrm{x}} p + \frac{1}{2} \nabla_{\mathrm{x}} \cdot \left(\mathrm{D}_f \nabla_{\mathrm{x}} p\right) + \left(-\nabla_{\mathrm{x}} \cdot \mu_f\right) \cdot p, \\ -\frac{\partial}{\partial \tau} \hat{p} &= \quad \mu_r^T \nabla_{\bar{\mathrm{x}}} \hat{p} + \frac{1}{2} \nabla_{\bar{\mathrm{x}}} \cdot \left(\mathrm{D}_r \nabla_{\bar{\mathrm{x}}} \hat{p}\right) - \left(-\nabla_{\bar{\mathrm{x}}} \cdot \mu_r\right) \cdot \hat{p}, \\ -\frac{\partial}{\partial \tau} \hat{p} &= \quad \mu_r^T \nabla_{\bar{\mathrm{x}}} \hat{p} + \frac{1}{2} \nabla_{\bar{\mathrm{x}}} \cdot \left(\mathrm{D}_r \nabla_{\bar{\mathrm{x}}} \hat{p}\right) - v_r \cdot \hat{p}, \ \tau \leq T, \\ & \quad \hat{p}\left(\bar{\mathrm{x}}_{\tau=T}, T\right) = \hat{p}_T\left(\bar{\mathrm{x}}_T\right) = p_t\left(\mathrm{x}_t\right), \bar{\mathrm{x}}_{\tau} \in \mathbb{R}^{d_x}; \end{aligned} \tag{2.154}$$

where

$$v_r\left(\bar{\mathrm{x}}_{\tau}, \tau\right) = -\nabla_{\bar{\mathrm{x}}} \mu_r\left(\bar{\mathrm{x}}_{\tau}, \tau\right). \tag{2.155}$$

Solving (2.154) corresponds to the Cauchy problem in reverse time $\tau \leq T$, which is equivalent to solve the stochastic system stated by (2.149) and (2.150). Therefore, because the solution to the SDE (2.148) is assumed to exist and corresponds to the solution of (2.154) for $\tau \leq T$, then there exists the equivalent reverse process $\left\{\bar{\mathrm{X}}_{\tau}, \mathcal{V}_{\tau}\right\}_{\tau \in [t,T]}$ that solves the stochastic system (2.149) and (2.150). □

Remark 2.14. Despite its name, inherited from [140], it is worth stressing that $\left\{\bar{\mathrm{X}}_{\tau}, \mathcal{V}_{\tau}\right\}$ is the solution of a stochastic system forward in time on the interval $[t, T]$, which may be properly understood as a smoothing process.

Remark 2.15. It is clear that the solution of (2.150) at $\tau = T$ is

$$\mathcal{V}_T \equiv \mathcal{V}_{\tau}\left(T\right) = e^{-\int_t^T v_r\left(\bar{\mathrm{x}}_{\tau}, \tau\right) d\tau},$$

which evokes the solution of (2.154) by the Feynman-Kac formula

$$\hat{p}\left(\mathrm{x}, \tau\right) = \mathbb{E}^{\mathcal{P}}\left[e^{-\int_{\tau}^T v_r\left(\bar{\mathrm{x}}_{\tau'}, \tau'\right) d\tau'} \hat{p}_T\left(\bar{\mathrm{x}}_T\right) | \bar{\mathrm{x}}_{\tau} = \mathrm{x}\right]. \tag{2.156}$$

A more general form of the *Lemma 2.13* can be found in [140].

Lemma 2.16. *The reverse process* $\left\{\bar{\mathrm{X}}_{\tau}, \mathcal{V}_{\tau} : t \leq \tau \leq T\right\}$ *described by (2.149) and (2.150), has an associated smooth probability density* $\hat{p}\left(\bar{\mathrm{x}}_{\tau}, \tau\right)$ *that satisfies, for the initial value problem, the Kolmogorov forward equation*

$$\begin{aligned} \frac{\partial}{\partial \tau} \hat{p} &= -\nabla_{\bar{\mathrm{x}}} \cdot \left(\mu_r \hat{p}\right) + \frac{1}{2} \nabla_{\bar{\mathrm{x}}} \cdot \left(\mathrm{D}_r \nabla_{\bar{\mathrm{x}}} \hat{p}\right) - v_r \cdot \hat{p}, \ t \leq \tau \leq T, \\ & \quad \hat{p}\left(\bar{\mathrm{x}}_{\tau=t}, \tau = t\right) = \hat{p}_t\left(\bar{\mathrm{x}}_t\right), \bar{\mathrm{x}}_{\tau} \in \mathbb{R}^{d_x}. \end{aligned} \tag{2.157}$$

Proof. We consider a continuous function of the process $\left\{\bar{\mathrm{X}}_\tau, \mathcal{V}_\tau\right\}_{\tau\in[t,T]}$, declared as $\varphi : \mathbb{R}^{d_x} \times \mathbb{R} \to \mathcal{X}'$, which is assumed to be $\varphi(\bar{\mathrm{X}}_\tau, \mathcal{V}_\tau) \in \mathcal{C}^2(\mathbb{R}^{d_x}, \mathbb{R})$, bounded and integrable on the product space $\mathbb{R}^{d_x} \times \mathbb{R}$. Applying Îto's lemma to φ and substituting (2.149) and (2.150) one obtains

$$\begin{aligned}
d\varphi &= (\nabla_{\bar{\mathrm{X}}_\tau}\varphi)^T d\bar{\mathrm{X}}_\tau + \frac{1}{2} d\bar{\mathrm{X}}_\tau^T \mathcal{H}_{\bar{\mathrm{X}}_\tau}[\varphi]\, d\bar{\mathrm{X}}_\tau + \partial_{\mathcal{V}_\tau}\varphi\, d\mathcal{V}_\tau \\
&= (\nabla_{\bar{\mathrm{X}}_\tau}\varphi)^T \cdot \left(\mu_r d\tau + D_r{}^{1/2} d\bar{\mathrm{W}}_\tau\right) + \frac{1}{2}\mathrm{tr}\left\{\mathrm{D}_r \mathcal{H}_{\bar{\mathrm{X}}_\tau}[\varphi]\right\} d\tau + \partial_{\mathcal{V}_\tau}\varphi \cdot (v_r \mathcal{V}_\tau d\tau), \\
&= \left[(\nabla_{\bar{\mathrm{X}}_\tau}\varphi)^T \mu_r + \frac{1}{2}\mathrm{tr}\left\{\mathrm{D}_r \mathcal{H}_{\bar{\mathrm{X}}_\tau}[\varphi]\right\} + \partial_{\mathcal{V}_\tau}\varphi \cdot v_r \mathcal{V}_\tau\right] d\tau + (\nabla_{\bar{\mathrm{X}}_\tau}\varphi)^T \mathrm{D}_r{}^{1/2} d\bar{\mathrm{W}}_\tau. \qquad (2.158)
\end{aligned}$$

Consider the expected (average) rate of change of the projections of φ defined by:

$$\begin{aligned}
\langle \dot{\varphi}_{\mathcal{V}} \rangle (\tau) &\triangleq \left\langle \frac{d}{d\tau} \int_1^{\mathcal{V}_T} \varphi(\bar{\mathrm{X}}_\tau, \mathcal{V}_\tau) d\mathcal{V}_\tau \right\rangle \\
&= \left\langle \int_1^{\mathcal{V}_T} \partial_\tau \varphi(\bar{\mathrm{X}}_\tau, \mathcal{V}_\tau) d\mathcal{V}_\tau \right\rangle \\
&\equiv \int_{\mathbb{R}^{d_x}} \int_1^{\upsilon_T} \varphi(\bar{\mathrm{x}}_\tau, \upsilon_\tau) \cdot \partial_\tau \hat{p}\, d\upsilon_\tau d\bar{\mathrm{x}}_\tau. \qquad (2.159)
\end{aligned}$$

Substituting (2.158) into (2.159), we have

$$\begin{aligned}
\langle \dot{\varphi}_{\mathcal{V}} \rangle (\tau) &= \left\langle \int_1^{\mathcal{V}_T} \partial_\tau \varphi(\bar{\mathrm{X}}_\tau, \mathcal{V}_\tau) d\mathcal{V}_\tau \right\rangle \\
&= \left\langle \int_1^{\mathcal{V}_T} \left[(\nabla_{\bar{\mathrm{X}}_\tau}\varphi)^T \mu_r + \frac{1}{2}\mathrm{tr}\left\{\mathrm{D}_r \mathcal{H}_{\bar{\mathrm{X}}_\tau}[\varphi]\right\} + \partial_{\mathcal{V}_\tau}\varphi \cdot v_r \mathcal{V}_\tau\right] d\mathcal{V}_\tau \right\rangle \\
&= \int_1^{\upsilon_T} \int_{\mathbb{R}^{d_x}} \left[(\nabla_{\bar{\mathrm{x}}_\tau}\varphi)^T \mu_r + \frac{1}{2}\mathrm{tr}\left\{\mathrm{D}_r \mathcal{H}_{\bar{\mathrm{x}}_\tau}[\varphi]\right\}\right] \hat{p}\, d\bar{\mathrm{x}}_\tau d\upsilon_\tau \\
&+ \int_{\mathbb{R}^{d_x}} \int_1^{\upsilon_T} \left[\partial_{\upsilon_\tau}\varphi \cdot v_r\, \upsilon_\tau\right] \hat{p}\, d\upsilon_\tau d\bar{\mathrm{x}}_\tau. \qquad (2.160)
\end{aligned}$$

Integrating (2.160) by parts and offsetting the integration constant to cancel out the surface terms, we have

$$\begin{aligned}
\langle \dot{\varphi}_{\mathcal{V}} \rangle (\tau) &= \int_1^{\upsilon_T} \int_{\mathbb{R}^{d_x}} \varphi \cdot \left[-\nabla_{\bar{\mathrm{x}}_\tau} \cdot (\mu_r \hat{p}) + \frac{1}{2}\nabla_{\bar{\mathrm{x}}_\tau} \cdot (\mathrm{D}_r \nabla_{\bar{\mathrm{x}}_\tau} \hat{p})\right] d\bar{\mathrm{x}}_\tau d\upsilon_\tau \\
&+ \int_{\mathbb{R}^{d_x}} \int_1^{\upsilon_T} \varphi \cdot \left[-v_r \partial_{\upsilon_\tau}(\upsilon_\tau \hat{p})\right] d\upsilon_\tau d\bar{\mathrm{x}}_\tau \\
&= \int_{\mathbb{R}^{d_x}} \int_1^{\upsilon_T} \varphi \cdot \left[-\nabla_{\bar{\mathrm{x}}_\tau} \cdot (\mu_r \hat{p}) + \frac{1}{2}\nabla_{\bar{\mathrm{x}}_\tau} \cdot (\mathrm{D}_r \nabla_{\bar{\mathrm{x}}_\tau} \hat{p})\right] d\upsilon_\tau d\bar{\mathrm{x}}_\tau \\
&+ \int_{\mathbb{R}^{d_x}} \int_1^{\upsilon_T} \varphi \cdot \left[-v_r \hat{p}\right] d\upsilon_\tau d\bar{\mathrm{x}}_\tau \\
&= \int_{\mathbb{R}^{d_x}} \int_1^{\upsilon_T} \varphi \cdot \left[-\nabla_{\bar{\mathrm{x}}_\tau} \cdot (\mu_r \hat{p}) + \frac{1}{2}\nabla_{\bar{\mathrm{x}}_\tau} \cdot (\mathrm{D}_r \nabla_{\bar{\mathrm{x}}_\tau} \hat{p}) - v_r \hat{p}\right] d\upsilon_\tau d\bar{\mathrm{x}}_\tau. \qquad (2.161)
\end{aligned}$$

The proof of the lemma is complete by comparing (2.161) to the definition (2.159) and noting that their integrands must be equal. □

Theorem 2.17. *Let $(\Omega, \mathcal{F}, \mathcal{P})$ to be a complete probability space and let $\{\mathcal{F}_\lambda\}_{\lambda\geq 0}$, $\lambda \in [0,T]$, be an increasing family of sub σ-fields of $\mathcal{F}$. Let $\{\mathrm{X}_\lambda : 0 < \lambda \leq T\}$ be an $\mathcal{F}_\lambda$-adapted process, considered to be the signal process with state equation*

$$\begin{aligned} d\mathrm{X}_\lambda &= \mu_f\left(\mathrm{X}_\lambda\right) d\lambda + \mathrm{D}_f\left(\lambda\right)^{1/2} dW_\lambda, \\ \mathrm{X}_{\lambda=0} &= \mathrm{x}_0,\, 0 \leq \lambda \leq T; \end{aligned} \tag{2.162}$$

for a Wiener process $\{\mathrm{W}_\lambda\}_{\lambda\in[0,T]}$ under the probability measure $\mathcal{P}$. Assume $p(\mathrm{x}_\lambda, \lambda)$ defined on $(\Omega, \mathcal{F})$ to be the probability density of the measure $\mathcal{P}$, which

(a) is the probabilistic representation of the process $\{\mathrm{X}_\lambda\}_{\lambda\in[0,T]}$,

(b) is absolutely continuous with respect to the Lebesgue measure,

(c) approaches a stationary measure as $p(\mathrm{x}_\lambda) \overset{\lambda\to T}{\longrightarrow} \pi\left(\mathrm{x}_\lambda\right) \propto e^{-\Phi(\mathrm{x}_\lambda)}$, for a sufficiently long horizon T.

Let $\left\{\bar{\mathrm{X}}_\tau, \mathcal{V}_\tau : \lambda < \tau \leq T\right\}$ be the reverse process of $\{\mathrm{X}_\lambda\}_{\lambda\in[0,T]}$, as established in Lemma 2.13 by the stochastic system (2.149) and (2.150), so that the reverse drift and diffusion coefficients are, respectively,

$$\begin{aligned} \mu_r\left(\bar{\mathrm{x}}_\tau\right) &= -\frac{1}{2}\mathrm{D}_r\left(\lambda\right)\nabla_{\bar{\mathrm{x}}} \log \pi\left(\bar{\mathrm{x}}_\tau\right), \\ \mathrm{D}_r\left(\tau\right) &= \left[-\mathcal{H}_{\bar{\mathrm{x}}}\left[\log \pi\left(\bar{\mathrm{x}}_\tau\right)\right]\right]^{-1}. \end{aligned}$$

Assume $\hat{p}(\bar{\mathrm{x}}_\tau, \tau)$ to be the probability density on $(\Omega, \mathcal{F})$ describing the reverse process $\left\{\bar{\mathrm{X}}_\tau, \mathcal{V}_\tau\right\}_{\tau\in[\lambda,T]}$, under the same measure $\mathcal{P}$, which must satisfy the Kolmogorov forward equation (provided a known initial condition) according to Lemma 2.16:

$$\begin{aligned} \frac{\partial}{\partial\tau}\hat{p} &= -\nabla_{\bar{\mathrm{x}}}\cdot\left(\mu_r\hat{p}\right) + \frac{1}{2}\nabla_{\bar{\mathrm{x}}}\cdot\left(\mathrm{D}_r\nabla_{\bar{\mathrm{x}}}\hat{p}\right) - v_r\cdot\hat{p},\ \lambda \leq \tau \leq T, \\ \hat{p}\left(\bar{\mathrm{x}}_{\tau=\lambda}, \tau=\lambda\right) &= \hat{p}_\lambda\left(\bar{\mathrm{x}}_{\tau=\lambda}\right) = \pi\left(\mathrm{x}_T\right), \bar{\mathrm{x}}_\tau \in \mathbb{R}^{d_x}; \end{aligned} \tag{2.163}$$

If the stationary density is set to be

$$\pi\left(\mathrm{x}\right) \coloneqq \frac{p\left(\mathrm{y}_k|\mathrm{x}\right) p\left(\mathrm{x}|\mathrm{y}_{1:k-1}\right)}{Z_1} = \frac{p\left(\mathrm{y}_k|\mathrm{x}\right) p_x\left(\mathrm{x}\right)}{Z_1}, \tag{2.164}$$

where the prior density $p_x\left(\mathrm{x}\right)$ and the likelihood $p\left(\mathrm{y}_k|\mathrm{x}\right)$ are integrable functions with respect to x, $Z_1 = p\left(\mathrm{y}_k|\mathrm{y}_{1:k-1}\right)$ is a normalization constant, and the discrete-time observation process $\left\{\mathrm{y}_k \in \mathbb{R}^{d_y} : k \in \mathbb{N}\right\}$ is described as

$$\mathrm{y}_k = h\left(\mathrm{x}_k\right) + \mathrm{R}^{1/2}\xi_k, \quad \xi_k \sim \mathcal{N}(\xi_k; 0_{d_y}, \mathbb{I}_{d_y}); \tag{2.165}$$

then the probability density corresponding to the signal process (2.162) is equivalent to the following filtering entity

$$p(\mathrm{x}, \lambda|\mathcal{F}_\lambda) = \frac{\mathbb{E}^{\mathcal{P}}\left[e^{h(\mathrm{x}_T)^T\mathrm{R}^{-1}\mathrm{y}_k - \frac{1}{2}h(\mathrm{x}_T)^T\mathrm{R}^{-1}h(\mathrm{x}_T)}|\mathrm{x}\right] p\left(\mathrm{x}|\mathrm{y}_{1:k-1}\right)}{Z}. \tag{2.166}$$

In addition, expression (2.166) can be interpreted as analogous to the well known result [19]

$$p(\mathrm{x},\lambda|\mathcal{F}_\lambda)=\frac{\mathbb{E}^{\mathcal{P}}\left[e^{\int_0^T h_T(\mathrm{x}_\lambda)^T\mathrm{R}_T^{-1}d\mathrm{y}_\lambda-\frac{1}{2}\int_0^T h_T(\mathrm{x}_\lambda)^T\mathrm{R}_T^{-1}h_T(\mathrm{x}_\lambda)d\lambda}|\mathrm{x}\right]}{Z}p\left(\mathrm{x}|\mathrm{y}_{1:k-1}\right), \tag{2.167}$$

for a discrete-time observation process whose analog continuous-time (interpolated) version has the observation function $h_T(.)$ and covariance matrix R_T.

Proof. By definition of the stochastic system described by (2.149) and (2.150), and *Lemma 2.16*, the reverse process $\left\{\bar{\mathrm{X}}_\tau,\mathcal{V}_\tau\right\}$ is known to satisfy the Kolmogorov forward equation (2.163) in reverse time $\lambda\leq\tau\leq T$, for which the stationary density π is an initial condition (initial value problem). Using the reverse time variable $\tau=T-\lambda$, so that $p\left(\mathrm{x}_\lambda,\lambda\right)\equiv\hat{p}\left(\bar{\mathrm{x}}_{T-\tau},T-\tau\right)$ and $\partial_\lambda p=-\partial_\tau\hat{p}$, and applying the relations (2.151), (2.152) and (2.155) from *Lemma 2.13*, we rewrite the equation (2.163) for $0\leq\lambda\leq T$ as

$$\begin{aligned}
\frac{\partial}{\partial\tau}\hat{p}&=-\nabla_{\bar{\mathrm{x}}}\cdot\left(\mu_r\hat{p}\right)+\frac{1}{2}\nabla_{\bar{\mathrm{x}}}\cdot\left(\mathrm{D}_r\nabla_{\bar{\mathrm{x}}}\hat{p}\right)-v_r\cdot\hat{p},\ \lambda\leq\tau\leq T,\\
\frac{\partial}{\partial\tau}\hat{p}&=-\left(\nabla_{\bar{\mathrm{x}}}\cdot\mu_r\right)\hat{p}-\mu_r^T\nabla_{\bar{\mathrm{x}}}\hat{p}+\frac{1}{2}\nabla_{\bar{\mathrm{x}}}\cdot\left(\mathrm{D}_r\nabla_{\bar{\mathrm{x}}}\hat{p}\right)+\left(\nabla_{\bar{\mathrm{x}}}\cdot\mu_r\right)\hat{p},\\
\frac{\partial}{\partial\tau}\hat{p}&=-\mu_r^T\nabla_{\bar{\mathrm{x}}}\hat{p}+\frac{1}{2}\nabla_{\bar{\mathrm{x}}}\cdot\left(\mathrm{D}_r\nabla_{\bar{\mathrm{x}}}\hat{p}\right),\\
-\frac{\partial}{\partial\lambda}p&=+\mu_f^T\nabla_{\mathrm{x}}p+\frac{1}{2}\nabla_{\mathrm{x}}\cdot\left(\mathrm{D}_f\nabla_{\mathrm{x}}p\right),\ 0\leq\lambda\leq T,\\
p\left(\mathrm{x}_\lambda,T\right)&=\pi\left(\mathrm{x}_T\right),\mathrm{x}_\lambda\in\mathbb{R}^{d_x}.
\end{aligned} \tag{2.168}$$

Now we have a Kolmogorov backward equation in $p\left(\mathrm{x}_\lambda,\lambda\right)$ with a terminal value problem for the forward process $\left\{\mathrm{X}_\lambda\right\}_{\lambda\in[0,T]}$. Hence, we can apply the Feynman-Kac formula for the terminal condition $p(\mathrm{x},T)=\pi\left(\mathrm{x}\right)$, with $\mathrm{x}_T=\mathrm{x}_{\lambda=T}$, to give

$$\begin{aligned}
p(\mathrm{x},\lambda)&\triangleq\mathbb{E}^{\mathcal{P}}\left[\pi\left(\mathrm{x}_T\right)|\mathrm{x}_\lambda=\mathrm{x}\right]\\
&=\frac{\mathbb{E}^{\mathcal{P}}\left[p\left(\mathrm{y}_k|\mathrm{x}_T\right)p\left(\mathrm{x}_T|\mathrm{y}_{1:k-1}\right)|\mathrm{x}_\lambda=\mathrm{x}\right]}{Z_1}\\
&=\frac{\mathbb{E}^{\mathcal{P}}\left[e^{-\frac{1}{2}\left(\mathrm{y}_k-h(\mathrm{x}_T)\right)^T\mathrm{R}^{-1}\left(\mathrm{y}_k-h(\mathrm{x}_T)\right)}p_x\left(\mathrm{x}_T\right)|\mathrm{x}\right]}{Z_1\left(2\pi\mathrm{R}\right)^{d_y/2}}\\
&=\frac{\mathbb{E}^{\mathcal{P}}\left[e^{h(\mathrm{x}_T)^T\mathrm{R}^{-1}\mathrm{y}_k-\frac{1}{2}h(\mathrm{x}_T)^T\mathrm{R}^{-1}h(\mathrm{x}_T)}p_x\left(\mathrm{x}_T\right)|\mathrm{x}\right]}{Z_1\left(2\pi\mathrm{R}\right)^{d_y/2}e^{+\frac{1}{2}\mathrm{y}_k^T\mathrm{R}^{-1}\mathrm{y}_k}}\\
&=\frac{\mathbb{E}^{\mathcal{P}}\left[e^{h(\mathrm{x}_T)^T\mathrm{R}^{-1}\mathrm{y}_k-\frac{1}{2}h(\mathrm{x}_T)^T\mathrm{R}^{-1}h(\mathrm{x}_T)}p_x\left(\mathrm{x}_T\right)|\mathcal{F}_\lambda,\mathrm{x}\right]}{Z}.
\end{aligned} \tag{2.169}$$

Let us reinterpret the discrete-time observation process as a continuous-time process for which we only obtain a realization at $\lambda=T$, by linearly interpolating it along the interval $0<\lambda\leq T$ to write

$$\begin{aligned}
d\mathrm{y}_\lambda&=\frac{1}{T}h\left(\mathrm{x}_\lambda\right)d\lambda+\left(\frac{\mathrm{R}}{T}\right)^{1/2}\xi_k d\lambda^{1/2},\\
d\mathrm{y}_\lambda^{\mathcal{Q}}&=h_T\left(\mathrm{x}_\lambda\right)d\lambda+\mathrm{R}_T^{1/2}d\bar{\xi}_\lambda^{\mathcal{P}},\quad 0<\lambda\leq T.
\end{aligned} \tag{2.170}$$

where $\left\{\bar{\xi}_\lambda\right\}$ is an interpolated Wiener process, under the probability measure $\mathcal{P}$, that produces

the observation noise $\int_0^T \mathrm{R}_T^{1/2} d\bar{\xi}_\lambda \equiv \mathrm{R}^{1/2}\xi_k$ at $\lambda = T$. The probability measure $\mathcal{P}$ is induced in the space of paths jointly described by the state process and observation noise $(\{\mathrm{X}_\lambda\}, \{\bar{\Xi}_\lambda\})$. The interpolation is established such that $\mathrm{y}_{\lambda=T} = \mathrm{y}_k$ is the realization of the observation process under the probability measure $\mathcal{Q}$, which is induced in the space of paths jointly described by the state and observation processes $(\{\mathrm{X}_\lambda\}, \{\mathrm{Y}_\lambda\})$. By applying the Girsanov's theorem, we know that the Radon-Nykodym derivative to change the measure from $\mathcal{P}$ to $\mathcal{Q}$ assumes the form (see [211] for example)

$$\begin{aligned}\left.\frac{d\mathcal{Q}}{d\mathcal{P}}\right|_{\mathcal{F}_T} &= e^{\int_0^T h_T(\mathrm{x}_\lambda)^T \mathrm{R}_T^{-1} d\mathrm{y}_\lambda - \frac{1}{2}\int_0^T h_T(\mathrm{x}_\lambda)^T \mathrm{R}_T^{-1} h_T(\mathrm{x}_\lambda) d\lambda} \\ &\propto e^{h(\mathrm{x}_T)^T \mathrm{R}^{-1}\mathrm{y}_k - \frac{1}{2}h(\mathrm{x}_T)^T \mathrm{R}^{-1} h(\mathrm{x}_T)}. \end{aligned} \tag{2.171}$$

Rewriting (2.169) in terms of (2.171) and manipulating it further, we obtain

$$\begin{aligned} p(\mathrm{x},\lambda) &\propto \mathbb{E}^{\mathcal{P}}\left[e^{h(\mathrm{x}_T)^T \mathrm{R}^{-1}\mathrm{y}_k - \frac{1}{2}h(\mathrm{x}_T)^T \mathrm{R}^{-1} h(\mathrm{x}_T)} p_x\left(\mathrm{x}_T\right)|\mathcal{F}_\lambda, \mathrm{x}\right] \\ &\equiv \mathbb{E}^{\mathcal{P}}\left[\frac{d\mathcal{Q}}{d\mathcal{P}}\left(T\right)\cdot p_x\left(\mathrm{x}_T\right)|\mathcal{F}_\lambda, \mathrm{x}\right] \\ &= \mathbb{E}^{\mathcal{P}}\left[\frac{d\mathcal{Q}}{d\mathcal{P}}\left(T\right)\cdot \mathbb{E}^{\mathcal{Q}}\left[p_x\left(\mathrm{x}_\lambda\right)|\mathcal{F}_T, \mathrm{x}\right]|\mathcal{F}_\lambda, \mathrm{x}\right] \\ &= \mathbb{E}^{\mathcal{P}}\left[\frac{d\mathcal{Q}}{d\mathcal{P}}\left(T\right)|\mathcal{F}_\lambda, \mathrm{x}\right]\cdot \mathbb{E}^{\mathcal{Q}}\left[p_x\left(\mathrm{x}_\lambda\right)|\mathcal{F}_T, \mathrm{x}\right] \\ &\equiv \mathbb{E}^{\mathcal{P}}\left[e^{h(\mathrm{x}_T)^T \mathrm{R}^{-1}\mathrm{y}_k - \frac{1}{2}h(\mathrm{x}_T)^T \mathrm{R}^{-1} h(\mathrm{x}_T)}|\mathcal{F}_\lambda, \mathrm{x}\right] p_x\left(\mathrm{x}\right), \end{aligned} \tag{2.172}$$

where we take into account the smoothing property for conditional expectations as

$$\begin{aligned} p(\mathrm{x},\lambda) &\propto \mathbb{E}^{\mathcal{P}}\left[\frac{d\mathcal{Q}}{d\mathcal{P}}\left(T\right)\cdot p_x\left(\mathrm{x}_T\right)|\mathcal{F}_\lambda, \mathrm{x}_\lambda = \mathrm{x}\right] \\ &= \mathbb{E}^{\mathcal{Q}}\left[p_x\left(\mathrm{x}_T\right)|\mathcal{F}_\lambda, \mathrm{x}\right] \\ &= \mathbb{E}^{\mathcal{Q}}\left[\mathbb{E}^{\mathcal{Q}}\left[p_x\left(\mathrm{x}_\lambda\right)|\mathcal{F}_T, \mathrm{x}\right]|\mathcal{F}_\lambda, \mathrm{x}\right] \\ &= \mathbb{E}^{\mathcal{P}}\left[\frac{d\mathcal{Q}}{d\mathcal{P}}\left(T\right)\cdot \mathbb{E}^{\mathcal{Q}}\left[p_x\left(\mathrm{x}_\lambda\right)|\mathcal{F}_T, \mathrm{x}\right]|\mathcal{F}_\lambda, \mathrm{x}\right], \end{aligned}$$

the $\mathcal{F}_\lambda$-measurability of $\mathbb{E}^{\mathcal{Q}}\left[p_x\left(\mathrm{x}_\lambda\right)|\mathcal{F}_T, \mathrm{x}_\lambda = \mathrm{x}\right]$, and

$$\begin{aligned}\mathbb{E}^{\mathcal{Q}}\left[p_x\left(\mathrm{x}_\lambda\right)|\mathcal{F}_T, \mathrm{x}_\lambda = \mathrm{x}\right] &= \int p_x\left(\mathrm{x}\right) d\mathcal{Q} \\ &= p_x\left(\mathrm{x}\right) = p\left(\mathrm{x}|\mathrm{y}_{1:k-1}\right). \end{aligned}$$

As a result, the expression (2.172) can be written in the normalized form (2.166), proving the theorem statement. The proof is complete by inserting the continuous-time (interpolated) version of (2.171) into (2.172) to verify the analogy with (2.167). □

Remark 2.18. A result more general than the one presented by *Theorem 2.17*, in terms of McKean-Vlasov diffusions, can be found in [39].

Corollary 2.19. *The signal process with state equation (2.162), under the hypotheses of Theorem 2.17, filters its associated (unnormalized) probability density in accordance with the Zakai equation*

$$dp_u = \mathcal{L}\left[p_u\right] d\lambda + p_u \cdot h_T\left(\mathrm{x}_\lambda\right)^T \mathrm{R}_T^{-1} d\mathrm{y}_\lambda, \quad 0 < \lambda \leq T; \tag{2.173}$$

where $\mathcal{L}\left[.\right] = -\nabla_{\mathrm{x}} \cdot (\mu \cdot) + {}^1\!/_2 \nabla_{\mathrm{x}} \cdot (\mathrm{D} \nabla_{\mathrm{x}} (\cdot))$ *is the forward Kolmogorov operator, and* $\{\mathrm{y}_\lambda : 0 < \lambda \leq T\}$ *is the continuous, linearly interpolated observation process defined by (2.170) for which the realization is only taken at* $\lambda = T$*.*

Proof. Define

$$\begin{aligned} d\zeta_\lambda &= h_T\left(\mathrm{x}_\lambda\right)^T \mathrm{R}_T^{-1} d\mathrm{y}_\lambda - \frac{1}{2} h_T\left(\mathrm{x}_\lambda\right)^T \mathrm{R}_T^{-1} h_T\left(\mathrm{x}_\lambda\right) d\lambda \\ &= \frac{1}{2} h_T\left(\mathrm{x}_\lambda\right)^T \mathrm{R}_T^{-1} h_T\left(\mathrm{x}_\lambda\right) d\lambda + h_T\left(\mathrm{x}_\lambda\right)^T \mathrm{R}_T^{-1/2} d\bar{\xi}_\lambda, \end{aligned} \tag{2.174}$$

and recognize the unnormalized probability density to be the numerator of (2.167):

$$p_u = \mathbb{E}^{\mathcal{P}}\left[e^{\zeta_T} | \mathrm{x}\right] p_x\left(\mathrm{x}\right). \tag{2.175}$$

Applying Îto's Lemma to p_u we get

$$\begin{aligned} dp_u &= \partial_\lambda p_u d\lambda + \partial_\zeta p_u d\zeta_\lambda \\ &\quad + \frac{1}{2}\left[h_T\left(\mathrm{x}_\lambda\right)^T \mathrm{R}_T^{-1/2}\right]\left[\mathrm{R}_T^{-1/2} h_T\left(\mathrm{x}_\lambda\right)\right] \partial^2_{\zeta\zeta} p_u d\lambda \\ &= \partial_\lambda p_u d\lambda + \partial_\zeta p_u d\zeta_\lambda \\ &\quad + \frac{1}{2} h_T\left(\mathrm{x}_\lambda\right)^T \mathrm{R}_T^{-1} h_T\left(\mathrm{x}_\lambda\right) \partial^2_{\zeta\zeta} p_u d\lambda. \end{aligned} \tag{2.176}$$

Because

$$\begin{aligned} \partial_\zeta p_u &= \partial^2_{\zeta\zeta} p_u = p_u, \\ \partial_\lambda p_u &= \mathbb{E}^{\mathcal{P}}\left[e^{-\int_0^T d\zeta_\lambda} | \mathrm{x}\right] \partial_\lambda p_x\left(\mathrm{x}\right) = \mathcal{L}\left[p_u\right]; \end{aligned}$$

the expression (2.176) becomes the Zakai equation as

$$\begin{aligned} dp_u &= \mathcal{L}\left[p_u\right] d\lambda \\ &\quad + p_u \cdot \left[\frac{1}{2} h_T\left(\mathrm{x}_\lambda\right)^T \mathrm{R}_T^{-1} h_T\left(\mathrm{x}_\lambda\right) d\lambda + h_T\left(\mathrm{x}_\lambda\right)^T \mathrm{R}_T^{-1/2} d\bar{\xi}_\lambda\right] \\ &\quad + p_u \cdot \frac{1}{2} h_T\left(\mathrm{x}_\lambda\right)^T \mathrm{R}_T^{-1} h_T\left(\mathrm{x}_\lambda\right) d\lambda \\ &= \mathcal{L}\left[p_u\right] d\lambda \\ &\quad + p_u \cdot \left[h_T\left(\mathrm{x}_\lambda\right)^T \mathrm{R}_T^{-1} h_T\left(\mathrm{x}_\lambda\right) d\lambda + h_T\left(\mathrm{x}_\lambda\right)^T \mathrm{R}_T^{-1/2} d\bar{\xi}_\lambda\right] \\ &= \mathcal{L}\left[p_u\right] d\lambda + p_u \cdot h_T\left(\mathrm{x}_\lambda\right)^T \mathrm{R}_T^{-1}\left[h_T\left(\mathrm{x}_\lambda\right) d\lambda + \mathrm{R}_T^{1/2} d\bar{\xi}_\lambda\right] \\ &= \mathcal{L}\left[p_u\right] d\lambda + p_u \cdot h_T\left(\mathrm{x}_\lambda\right)^T \mathrm{R}_T^{-1} d\mathrm{y}_\lambda. \end{aligned}$$

□

2.9.3 Derivation of the integration rule

The method used in this section to derive the discrete-time integration rule is fully justified in Appendix A. We intend to approximate the integration of the following equation with respect to λ:

$$d\mathrm{x} = \frac{1}{2}\mathrm{D}\left(\lambda\right) \cdot \nabla_{\mathrm{x}} \log \pi\left(\mathrm{x}\right) d\lambda + \mathrm{D}\left(\lambda\right)^{1/2} d\mathrm{w}_{\lambda}. \tag{2.177}$$

Linearizing equation (2.177) w.r.t. x around the current state x_{n-1}, we have

$$d\mathrm{x} = \mathrm{A} \cdot \mathrm{x}\, d\lambda + \mathrm{B}\, d\lambda + \mathrm{D}^{1/2} d\mathrm{w}_{\lambda}, \tag{2.178}$$

where

$$\begin{aligned} \mathrm{A}(\mathrm{x}_{n-1}) &= {}^{1}\!/_{2}\mathrm{D}(\mathrm{x}_{n-1}) \cdot \mathcal{H}_{\mathrm{x}} \left[\log \pi\left(\mathrm{x}\right)\right]_{\mathrm{x}_{n-1}}, \\ \mathrm{B}(\mathrm{x}_{n-1}) &= \mathrm{a}(\mathrm{x}_{n-1}) - \mathrm{A} \cdot \mathrm{x}_{n-1}, \\ \mathrm{a}(\mathrm{x}_{n-1}) &= {}^{1}\!/_{2}\mathrm{D}(\mathrm{x}_{n-1}) \cdot \nabla_{\mathrm{x}} \log \pi\left(\mathrm{x}\right)\big|_{\mathrm{x}_{n-1}}. \end{aligned} \tag{2.179}$$

If we apply the definition $\mathrm{D}(\lambda_{n-1}) = -\mathcal{H}_{\mathrm{x}} \left[\log \pi\left(\mathrm{x}\right)\right]^{-1}_{\mathrm{x}_{n-1}}$, where $\mathcal{H}_{\mathrm{x}}\left[\cdot\right]$ is the Hessian w.r.t. x, we have

$$\mathrm{A} = -\frac{1}{2}\mathbb{I}_{d_x}, \tag{2.180}$$

where $\mathbb{I}_{d_x}$ is the identity matrix with dimension $d_x \times d_x$. Based on the Laplace transform, we can obtain the solution for a homogeneous version of the equation (2.178) in discrete time by

$$\begin{aligned} \mathrm{x}\left(\lambda\right) &= \mathcal{L}^{-1}\left\{(s \cdot \mathbb{I}_{d_x} - \mathrm{A})^{-1}\mathrm{x}(\lambda_{n-1})\right\} \\ &= \int_{\lambda_{n-1}}^{\lambda} (s \cdot \mathbb{I}_{d_x} - \mathrm{A})^{-1} e^{s \cdot \mathbb{I}_{d_x} \cdot \tau} \mathrm{x}(\lambda_{n-1}) ds \\ &= e^{\mathrm{A} \cdot (\lambda - \lambda_{n-1})} \mathrm{x}(\lambda_{n-1}), \\ \mathrm{x}_n &= e^{\mathrm{A} \cdot \Delta\lambda} \mathrm{x}_{n-1}. \end{aligned} \tag{2.181}$$

By a similar procedure, and considering the definition of a Wiener integral for the stochastic term, we can obtain the solution of the complete inhomogeneous equation (2.178) as

$$\begin{aligned} \mathrm{x}(\lambda_n) &= e^{\mathrm{A} \cdot \Delta\lambda} \mathrm{x}(\lambda_{n-1}) + \int_{\lambda_{n-1}}^{\lambda_n} e^{-\mathrm{A} \cdot (\tau - \Delta\lambda)} \mathrm{B}\, d\tau \\ &\quad + \sqrt{\int_{\lambda_{n-1}}^{\lambda_n} e^{-\mathrm{A} \cdot (\tau - \Delta\lambda)} \mathrm{D}(\lambda_{n-1}) e^{-\mathrm{A}^T \cdot (\tau - \Delta\lambda)} d\tau} \cdot \mathrm{w}_n, \\ \mathrm{x}_n &= e^{\mathrm{A} \cdot \Delta\lambda} \mathrm{x}_{n-1} + \mathrm{A}^{-1} \left[e^{\mathrm{A} \cdot \Delta\lambda} - \mathbb{I}_{d_x}\right] \left[\mathrm{a} - \mathrm{A} \cdot \mathrm{x}_{n-1}\right] \\ &\quad + \sqrt{\int_{0}^{\Delta\lambda} e^{\mathrm{A} \cdot \nu} \mathrm{D}(\lambda_{n-1}) e^{\mathrm{A}^T \cdot \nu} d\nu} \cdot \mathrm{w}_n \\ &= \mathrm{x}_{n-1} + \mathrm{A}^{-1} \left[e^{\mathrm{A} \cdot \Delta\lambda} - \mathbb{I}_{d_x}\right] \mathrm{a}(\lambda_{n-1}) \\ &\quad + \sqrt{\int_{0}^{\Delta\lambda} e^{\mathrm{A} \cdot \nu} \mathrm{D}(\lambda_{n-1}) e^{\mathrm{A}^T \cdot \nu} d\nu} \cdot \mathrm{w}_n, \end{aligned} \tag{2.182}$$

where $\mathrm{w}_n \sim \mathcal{N}(\mathrm{w};\, 0_{d_x}, \mathbb{I}_{d_x})$. Substituting (2.180) into (2.182), we have

$$\begin{aligned}\mathrm{x}_n = \mathrm{x}_{n-1} &+ \left[-\frac{1}{2}\mathbb{I}_{d_x}\right]^{-1}\left[e^{-\frac{1}{2}\mathbb{I}_{d_x}\cdot\Delta\lambda} - \mathbb{I}_{d_x}\right]\mathrm{a}(\mathrm{x}_{n-1}) \\ &+ \sqrt{\int_0^{\Delta\lambda} e^{-\frac{1}{2}\mathbb{I}_{d_x}\cdot\nu}\mathrm{D}(\lambda_{n-1})e^{-\frac{1}{2}\mathbb{I}_{d_x}^T\cdot\nu}d\nu}\cdot\mathrm{w}_n. \end{aligned} \tag{2.183}$$

By noticing that

$$e^{-\frac{1}{2}\mathbb{I}_{d_x}\Delta\lambda} = e^{-\frac{1}{2}\Delta\lambda}\mathbb{I}_{d_x}, \tag{2.184}$$

the equation (2.183) can be simplified as

$$\begin{aligned}\mathrm{x}_n &= \mathrm{x}_{n-1} - 2\left(e^{-\frac{\Delta\lambda}{2}} - 1\right)\mathrm{a}(\mathrm{x}_{n-1}) \\ &\quad + \sqrt{\int_0^{\Delta\lambda} e^{-\nu/2}e^{-\nu/2}d\nu}\cdot\mathrm{D}(\lambda_{n-1})^{1/2}\cdot\mathrm{w}_n \\ &= \mathrm{x}_{n-1} + 2\left(1 - e^{-\frac{\Delta\lambda}{2}}\right)\mathrm{a}(\mathrm{x}_{n-1}) \\ &\quad + \sqrt{\int_0^{\Delta\lambda} e^{-\nu}d\nu}\cdot\mathrm{D}(\lambda_{n-1})^{1/2}\cdot\mathrm{w}_n, \\ \mathrm{x}_n &= \mathrm{x}_{n-1} + \left(1 - e^{-\frac{\Delta\lambda}{2}}\right)\mathrm{D}(\lambda_{n-1})\cdot\nabla_{\mathrm{x}}\log\pi(\mathrm{x}_{n-1}) \\ &\quad + \left(1 - e^{-\Delta\lambda}\right)^{1/2}\mathrm{D}(\lambda_{n-1})^{1/2}\cdot\mathrm{w}_n. \end{aligned} \tag{2.185}$$

2.9.4 Justification for the local flow linearization

When approximating the stochastic particle flow (2.25) as locally linear in the neighborhood of a probability mass located at x_l, we expect to produce a negligible error in the propagated moments. Given a small increment of pseudo-time $\Delta\lambda > 0$, the SDE is approximated within the region $\|\mathrm{x} - \mathrm{x}_l\| < \zeta$, for a sufficiently small $\zeta \in \mathbb{R}_+$, as

$$d\mathrm{x} = \frac{1}{2}\mathrm{D}(\lambda)\nabla_{\mathrm{x}}\log\pi(\mathrm{x})\,d\lambda + \mathrm{D}(\lambda)^{1/2}d\mathrm{w}_\lambda, \quad \lambda \in (\lambda_l, \lambda_l + \Delta\lambda],\ \mathrm{x}(\lambda_l) = \mathrm{x}_l; \tag{2.186}$$

$$d\mathrm{x} \approx \left[\mathrm{C}(\mathrm{x}_l, \lambda)\cdot\mathrm{x} + \mathrm{c}(\mathrm{x}_l, \lambda)\right]d\lambda + \mathrm{D}(\lambda)^{1/2}d\mathrm{w}_\lambda. \tag{2.187}$$

In this section we provide a non-rigorous argument to explain why this local flow approximation produces admissible errors on the propagated moments without major concern. We will look at the expected error with respect to the intermediate marginal measures that follow from the Langevin dynamics for $\lambda \geq \lambda_l$ as

$$\begin{aligned} q(\mathrm{x}|\mathrm{y}_k) &= \int_{\mathcal{X}} p_t(\mathrm{x}|\mathrm{x}_l)p(\mathrm{x}_l|\mathrm{y}_k)d\mathrm{x}_l = \mathbb{E}_{p(l)}\left[p_t(\mathrm{x}|\mathrm{x}_l)\right] \\ &= \mathbb{E}_{p(l)}\left[\mathcal{N}\left(\mathrm{x};\, \mathrm{x}_l + \int_{\lambda_l}^{\lambda_l+\Delta\lambda}\mu(\mathrm{x}_l,\lambda)d\lambda,\, \left(\int_{\lambda_l}^{\lambda_l+\Delta\lambda}\mathrm{D}(\lambda)^{1/2}d\mathrm{w}_\lambda\right)^2\right)\right], \end{aligned}$$

where $\mu(\mathrm{x},\lambda) = \frac{1}{2}\mathrm{D}(\lambda)\nabla_{\mathrm{x}}\log\pi(\mathrm{x})$.

In the following analysis we will assume that the state is one-dimensional, i.e. $\mathrm{x} \in \mathbb{R}$, just to present a short argument that can be easily extended to the multidimensional case. If

one follows a procedure that (i) applies Îto's lemma to a continuous, measurable and nicely behaved function $\mathscr{M}(\mathrm{x})$, (ii) substitutes in the exact stochastic differential (2.186), (iii) takes expectation of the resulting equation, (iv) differentiates it with respect to λ, and then (v) integrates its right-hand side by parts, one obtains the so-called moment equation:

$$\frac{d}{d\lambda}\mathbb{E}_q\left[\mathscr{M}(\mathrm{x})\right] = \mathbb{E}_q\left[\frac{\partial \mathscr{M}(\mathrm{x})}{\partial \mathrm{x}}\mu(\mathrm{x},\lambda) + \frac{1}{2}\mathrm{D}(\lambda)\frac{\partial^2 \mathscr{M}(\mathrm{x})}{\partial \mathrm{x}^2}\right]. \tag{2.188}$$

Note that when applied to $\mathscr{M}(\mathrm{x}) = \mathrm{x}$ and $\mathscr{M}(\mathrm{x}) = (\mathrm{x}-\mu_m)^2$, for the linear approximation $\tilde{\mu}(\mathrm{x},\lambda) = \mathrm{C}(\lambda)\cdot\mathrm{x} + \mathrm{c}(\lambda)$, the referred equation gives the ODEs of the approximated mean and variance respectively

$$\begin{aligned}\frac{d}{d\lambda}\tilde{\mathbb{E}}_q\left[\mathrm{x}\right] &= \ \mathrm{C}(\lambda)\cdot\tilde{\mathbb{E}}_q\left[\mathrm{x}\right] + \mathrm{c}(\lambda),\\ \frac{d}{d\lambda}\tilde{\mathbb{E}}_q\left[(\mathrm{x}-\mu_m)^2\right] &= 2\mathrm{C}(\lambda)\cdot\tilde{\mathbb{E}}_q\left[(\mathrm{x}-\mu_m)^2\right] + \mathrm{D}(\lambda).\end{aligned}$$

Denote the deviation $\delta_{\mathbb{E}}\left[\mathrm{x}\right] = \mathbb{E}_q\left[\mathrm{x}\right] - \tilde{\mathbb{E}}_q\left[\mathrm{x}\right]$, where $\mathbb{E}_q\left[\mathrm{x}\right] = \mu_m$ is the exact mean propagated by the process (2.186) and $\tilde{\mathbb{E}}_q\left[\mathrm{x}\right] = \tilde{\mu}_m$ is the approximated mean propagated by the locally linearized process (2.187). First we note that the reverse triangle inequality allows us to state

$$\begin{aligned}\lim_{\Delta\lambda\to 0}\left|\frac{\|\delta_{\mathbb{E}}\left[\mathrm{x}(\lambda+\Delta\lambda)\right]\| - \|\delta_{\mathbb{E}}\left[\mathrm{x}(\lambda)\right]\|}{\Delta\lambda}\right| &\leq \lim_{\Delta\lambda\to 0}\frac{\|\delta_{\mathbb{E}}\left[\mathrm{x}(\lambda+\Delta\lambda)\right] - \delta_{\mathbb{E}}\left[\mathrm{x}(\lambda)\right]\|}{\Delta\lambda},\\ \left|\lim_{\Delta\lambda\to 0}\frac{\|\delta_{\mathbb{E}}\left[\mathrm{x}(\lambda+\Delta\lambda)\right]\| - \|\delta_{\mathbb{E}}\left[\mathrm{x}(\lambda)\right]\|}{\Delta\lambda}\right| &\leq \lim_{\Delta\lambda\to 0}\left\|\frac{\delta_{\mathbb{E}}\left[\mathrm{x}(\lambda+\Delta\lambda)\right] - \delta_{\mathbb{E}}\left[\mathrm{x}(\lambda)\right]}{\Delta\lambda}\right\|,\\ \therefore \left|\frac{d}{d\lambda}\left\|\delta_{\mathbb{E}}\left[\mathrm{x}(\lambda)\right]\right\|\right| &\leq \left\|\frac{d}{d\lambda}\delta_{\mathbb{E}}\left[\mathrm{x}(\lambda)\right]\right\|,\end{aligned} \tag{2.189}$$

where $|\cdot|$ is the absolute value, $\|\cdot\|$ the Euclidean norm, and all limits are assumed to exist. We use inequality (2.189) and the moment equation (2.188) to work out

$$\begin{aligned}\left|\frac{d}{d\lambda}\left\|\mathbb{E}_q\left[\mathrm{x}\right] - \tilde{\mathbb{E}}_q\left[\mathrm{x}\right]\right\|\right| &\leq \left\|\frac{d}{d\lambda}\mathbb{E}_q\left[\mathrm{x}\right] - \frac{d}{d\lambda}\tilde{\mathbb{E}}_q\left[\mathrm{x}\right]\right\|,\\ \left|\frac{d}{d\lambda}\left\|\mu_m - \tilde{\mu}_m\right\|\right| &\leq \left\|\mathbb{E}_q\left[\mu(\mathrm{x},\lambda)\right] - \mathbb{E}_q\left[\tilde{\mu}(\mathrm{x},\lambda)\right]\right\|, \quad \text{(moment equation for } \mathscr{M}(\mathrm{x}) = \mathrm{x})\\ \left|\frac{d}{d\lambda}\left\|\mu_m - \tilde{\mu}_m\right\|\right| &\leq \left\|\mathbb{E}_q\left[\mu(\mathrm{x},\lambda) - \tilde{\mu}(\mathrm{x},\lambda)\right]\right\| \leq \mathbb{E}_q\left[\left\|\mu(\mathrm{x},\lambda) - \tilde{\mu}(\mathrm{x},\lambda)\right\|\right],\\ \left|\frac{d}{d\lambda}\left\|\mu_m - \tilde{\mu}_m\right\|\right| &\leq \mathbb{E}_q\left[\left\|\mu(\mathrm{x},\lambda) - (\mathrm{C}(\mathrm{x}_l,\lambda)\cdot\mathrm{x} + \mathrm{c}(\mathrm{x}_l,\lambda))\right\|\right] \leq K_0\mathbb{E}_q\left[\left\|(\mathrm{x}-\mathrm{x}_l)^2\right\|\right],\\ \frac{d}{d\lambda}\left\|\mu_m - \tilde{\mu}_m\right\| &\leq K_1\zeta^2\\ \left\|\mu_m - \tilde{\mu}_m\right\| &\leq K_1\zeta^2\Delta\lambda, \quad \text{for } \lambda\in(\lambda_l,\lambda_l+\Delta\lambda],\ \mathrm{x}(\lambda_l) = \mathrm{x}_l,\ K_0, K_1\in\mathbb{R}_+,\end{aligned} \tag{2.190}$$

where the modulus is dismissed because $\|\mu_m - \tilde{\mu}_m\|$ increases monotonically with $\Delta\lambda$. Similarly

for the error on the second moment

$$
\begin{aligned}
\left|\frac{d}{d\lambda}\|\Sigma_m - \tilde{\Sigma}_m\|\right| &= \left\|\frac{d}{d\lambda}\mathbb{E}_q\left[(\mathrm{x}-\mu_m)^2\right] - \frac{d}{d\lambda}\tilde{\mathbb{E}}_q\left[(\mathrm{x}-\mu_m)^2\right]\right\|, \\
\left|\frac{d}{d\lambda}\|\Sigma_m - \tilde{\Sigma}_m\|\right| &\leq \left\|\mathbb{E}_q\left[2(\mathrm{x}-\mu_m)\mu(\mathrm{x},\lambda) + \mathrm{D}(\lambda)\right] - \mathbb{E}_q\left[2(\mathrm{x}-\mu_m)\tilde{\mu}(\mathrm{x},\lambda) + \mathrm{D}(\lambda)\right]\right\|, \\
\left|\frac{d}{d\lambda}\|\Sigma_m - \tilde{\Sigma}_m\|\right| &\leq \left\|2\mathbb{E}_q\left[(\mathrm{x}-\mu_m)(\mu(\mathrm{x},\lambda) - \tilde{\mu}(\mathrm{x},\lambda))\right]\right\| \leq 2\mathbb{E}_q\left[\left\|(\mathrm{x}-\mu_m)(\mu(\mathrm{x},\lambda) - \tilde{\mu}(\mathrm{x},\lambda))\right\|\right], \\
\left|\frac{d}{d\lambda}\|\Sigma_m - \tilde{\Sigma}_m\|\right| &\leq 2K_0'\mathbb{E}_q\left[\left\|(\mathrm{x}-\mu_m)(\mathrm{x}-\mathrm{x}_l)^2\right\|\right] = 2K_0'\mathbb{E}_q\left[\left\|(\mathrm{x}_l-\mu_m)(\mathrm{x}-\mathrm{x}_l)^2 + (\mathrm{x}-\mathrm{x}_l)^3\right\|\right], \\
\left|\frac{d}{d\lambda}\|\Sigma_m - \tilde{\Sigma}_m\|\right| &\leq 2K_0'\mathbb{E}_q\left[\left(\|\mathrm{x}_l-\mu_m\| \cdot \left\|(\mathrm{x}-\mathrm{x}_l)^2\right\| + \left\|(\mathrm{x}-\mathrm{x}_l)^3\right\|\right)\right], \text{ (triangle, Cauchy-Schwarz)} \\
\left|\frac{d}{d\lambda}\|\Sigma_m - \tilde{\Sigma}_m\|\right| &\leq K_{1a}'\zeta^2 + K_{1b}'\zeta^3, \\
\frac{d}{d\lambda}\|\Sigma_m - \tilde{\Sigma}_m\| &\leq K_1'\zeta^2, \\
\|\Sigma_m - \tilde{\Sigma}_m\| &\leq K_1'\zeta^2\Delta\lambda, \text{ for } \lambda \in (\lambda_l, \lambda_l + \Delta\lambda],\ K_0', K_{1a}', K_{1b}', K_1' \in \mathbb{R}^+,
\end{aligned}
\tag{2.191}
$$

where the modulus is suppressed because $\|\Sigma_m - \tilde{\Sigma}_m\|$ increases monotonically with $\Delta\lambda$.

It is very important to mention that the collection of factors K_0, K_1, K_0', K_{1a}', K_{1b}', K_1', can be different for each possible interval $(\lambda_l, \lambda_l + \Delta\lambda]$. Rigorously speaking those coefficients can depend on pseudo-time λ because

$$
K_0,\, K_0' \propto \frac{1}{2}\left\|\frac{\partial^2}{\partial \mathrm{x}^2}\mu(\mathrm{x},\lambda)\right\|_{\mathrm{x}=\mathrm{x}_l}.
$$

However, in accordance with the methodology of the stochastic particle flow, we select these factors so that inequalities (2.190) and (2.191) hold for a specific interval within which the diffusion coefficient is kept fixed as $\mathrm{D}(\lambda_l) \equiv \mathrm{D}(\mathrm{x}(\lambda_l)) = \mathrm{D}(\mathrm{x}_l)$. Given that the diffusion coefficient is piecewise constant in λ, the referred factors are also piecewise constant in λ and directly dependent on the diffusion coefficient. If the target density is Gaussian, then K_0, $K_0' = 0$ and the error committed due to the local linearization is null. For a fixed diffusion coefficient, K_0, $K_0' > 0$ if and only if the target log-density has third or higher-order non-zero derivatives at x_l.

Integrating (2.186) by the Euler-Maruyama scheme would produce $\mathbb{E}_q\left[\|\tilde{\mathrm{x}}_{l+1} - \mathrm{x}_{l+1}\|\right] \leq K\Delta\lambda^{1/2}$ and, therefore, it would be reasonable to expect $\|\mu_m - \tilde{\mu}_m\| \leq K_2\Delta\lambda^2$ and $\|\Sigma_m - \tilde{\Sigma}_m\| \leq K_2'\Delta\lambda^2$ for some $K_2, K_2' \in \mathbb{R}_+$. In the multivariate case, by applying the method used in this section, the curious reader should learn that the bounds are multiplied by the dimension d_x, to give $\|\mu_m - \tilde{\mu}_m\|_2 \leq K_1\zeta^2\Delta\lambda \cdot d_x$ and $\|\Sigma_m - \tilde{\Sigma}_m\|_2 \leq K_1'\zeta^2\Delta\lambda \cdot d_x$.

2.9.5 Discrete-time stochastic IDM

This section presents the resulting discrete-time approximation of the stochastic Intelligent Driver model. The method used for deriving the discrete-time IDM is explained later on in Appendix A. Define the state equation for the discrete-time IDM to be

$$
\mathrm{x}_k = \mathrm{A} \cdot \mathrm{x}_{k-1} + \mathrm{B} + \mathrm{w}_k, \tag{2.192}
$$

where the state vector is represented for α vehicles by

$$\mathrm{x}_k = \begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_{\alpha-1} \\ p_\alpha \\ v_1 \\ v_2 \\ \vdots \\ v_{\alpha-1} \\ v_\alpha \end{pmatrix}_k . \tag{2.193}$$

The variables p_i and v_i are the position and velocity of the ith vehicle respectively. The state-transition matrix can be written as

$$\mathrm{A} = \begin{pmatrix} \mathrm{A}_{1,1} & \mathrm{A}_{1,2} & \dots & \mathrm{A}_{1,2\alpha} \\ \mathrm{A}_{2,1} & \mathrm{A}_{2,2} & & \vdots \\ \vdots & & \ddots & \\ \mathrm{A}_{2\alpha,1} & \dots & & \mathrm{A}_{2\alpha,2\alpha} \end{pmatrix}_{k-1} . \tag{2.194}$$

For $i, j \in \mathbb{N}$, the diagonal elements of the state-transition matrix are given by

$$\mathrm{A}_{i,i} = \begin{cases} 1, & i \in [1, \alpha]; \\ 1 + \frac{\partial \dot{v}_{i-\alpha}}{\partial v_{i-\alpha}} \cdot \Delta t, & i \in (\alpha + 1, 2\alpha]; \end{cases} \tag{2.195}$$

and the off-diagonal elements given by

$$\mathrm{A}_{i,j} = \begin{cases} \Delta t, & i \in [1, \alpha],\ j = i + \alpha; \\ \frac{\partial \dot{v}_{i-\alpha}}{\partial p_\alpha} \cdot \Delta t, & i = \alpha + 1,\ j = \alpha; \\ \frac{\partial \dot{v}_{i-\alpha}}{\partial p_{i-\alpha-1}} \cdot \Delta t, & i \in (\alpha + 1, 2\alpha],\ j = i - 1 - \alpha; \\ \frac{\partial \dot{v}_{i-\alpha}}{\partial v_\alpha} \cdot \Delta t, & i = \alpha + 1,\ j = 2\alpha; \\ \frac{\partial \dot{v}_{i-\alpha}}{\partial v_{i-\alpha-1}} \cdot \Delta t, & i \in (\alpha + 1, 2\alpha],\ j = i - 1; \\ 0, & \text{otherwise}; \end{cases} \tag{2.196}$$

where

$$\frac{\partial \dot{v}_n}{\partial p_{n-1}} = +2a \left(\frac{s\left(v_n, \Delta v_n\right)^2}{s_n^3} \right), \tag{2.197}$$

$$\frac{\partial \dot{v}_n}{\partial p_n} = -2a \left(\frac{s\left(v_n, \Delta v_n\right)^2}{s_n^3} \right), \tag{2.198}$$

$$\frac{\partial \dot{v}_n}{\partial v_{n-1}} = +2a \left(\frac{s\left(v_n, \Delta v_n\right)}{s_n^2} \right) \left(\frac{v_n}{2\sqrt{a \cdot b}} \right), \tag{2.199}$$

$$\begin{aligned} \frac{\partial \dot{v}_n}{\partial v_n} &= -a \left(\frac{\delta}{v_0} \right) \left(\frac{v_n}{v_0} \right)^{\delta - 1} \\ &\quad - 2a \left(\frac{s\left(v_n, \Delta v_n\right)}{s_n^2} \right) \left(T_h + \frac{2v_n - v_{n-1}}{2\sqrt{a \cdot b}} \right). \end{aligned} \tag{2.200}$$

The model takes into account the fact that, on a ring road, the last vehicle in the convoy can be regarded the one potentially in front of the vehicle leading the queue, assuming that the first vehicle can complete the circuit faster and approach the last one from behind. This is represented by the terms $\partial_{p_\alpha} \dot{v}_1 \cdot \Delta t$ and $\partial_{v_\alpha} \dot{v}_1 \cdot \Delta t$ that appear in (2.196) when $i = \alpha + 1$, which shall be calculated respectively according to expressions analogous to (2.197) and (2.199). The constant term is defined as

$$\mathrm{B} = \begin{pmatrix} \mathrm{B}_1 \\ \mathrm{B}_2 \\ \vdots \\ \mathrm{B}_{2\alpha} \end{pmatrix}_{k-1}, \tag{2.201}$$

where

$$\mathrm{B}_i = \begin{cases} 0, & i \in [1, \alpha]; \\ \langle \dot{v}_i \rangle - \frac{\partial \dot{v}_i}{\partial p_{i-1}} \cdot p_{i-1} - \frac{\partial \dot{v}_i}{\partial p_i} \cdot p_i & \\ \quad - \frac{\partial \dot{v}_i}{\partial v_{i-1}} \cdot v_{i-1} - \frac{\partial \dot{v}_i}{\partial v_i} \cdot v_i, & i \in [\alpha + 1, 2\alpha]; \end{cases} \tag{2.202}$$

and

$$\langle \dot{v}_i \rangle = a \left[1 - \left(\frac{v_i}{v_0} \right)^\delta - \left(\frac{s\left(v_i, \Delta v_i\right)}{s_i} \right)^2 \right]. \tag{2.203}$$

The covariance matrix $\mathrm{Q}_k = \mathbb{E}\left[\mathrm{w}_k \mathrm{w}_k^T\right]$ is defined as

$$\mathrm{Q}_k = \begin{pmatrix} \mathrm{Q}_{1,1} & \mathrm{Q}_{1,2} & \cdots & \mathrm{Q}_{1,2\alpha} \\ \mathrm{Q}_{2,1} & \mathrm{Q}_{2,2} & & \vdots \\ \vdots & & \ddots & \\ \mathrm{Q}_{2\alpha,1} & \cdots & & \mathrm{Q}_{2\alpha,2\alpha} \end{pmatrix}_k, \tag{2.204}$$

with diagonal elements

$$\mathrm{Q}_{i,i} = \sigma_q^2 \times \begin{cases} \mathrm{Q}_{i,i}^{(1)}, & i \in [1, \alpha]; \\ \mathrm{Q}_{i,i}^{(2)}, & i = \alpha + 1; \\ \mathrm{Q}_{i,i}^{(3)}, & i \in (\alpha + 1, 2\alpha]; \end{cases} \tag{2.205}$$

where

$$\begin{aligned}
\mathrm{Q}_{i,i}^{(1)} &= \frac{\Delta t^3}{3} + \Delta t, \\
\mathrm{Q}_{i,i}^{(2)} &= \left(\frac{\partial \dot{v}_{i-\alpha}}{\partial p_\alpha} + \frac{\partial \dot{v}_{i-\alpha}}{\partial p_{i-\alpha}} + \frac{\partial \dot{v}_{i-\alpha}}{\partial v_\alpha} + \frac{\partial \dot{v}_{i-\alpha}}{\partial v_{i-\alpha}} \right) \cdot \frac{\Delta t^3}{3} \\
&+ \frac{\partial \dot{v}_{i-\alpha}}{\partial v_{i-\alpha}} \cdot \Delta t^2 + \Delta t, \\
\mathrm{Q}_{i,i}^{(3)} &= \left(\frac{\partial \dot{v}_{i-\alpha}}{\partial p_{i-\alpha-1}} + \frac{\partial \dot{v}_{i-\alpha}}{\partial p_{i-\alpha}} + \frac{\partial \dot{v}_{i-\alpha}}{\partial v_{i-\alpha-1}} + \frac{\partial \dot{v}_{i-\alpha}}{\partial v_{i-\alpha}} \right) \cdot \frac{\Delta t^3}{3} \\
&+ \frac{\partial \dot{v}_{i-\alpha}}{\partial v_{i-\alpha}} \cdot \Delta t^2 + \Delta t;
\end{aligned} \tag{2.206}$$

and off-diagonal elements

$$\mathrm{Q}_{j,i} = \mathrm{Q}_{i,j} = \sigma_q^2 \times \begin{cases} \mathrm{Q}_{i,j}^{(4)}, & i \in [1, \alpha],\ j = i + \alpha; \\ \mathrm{Q}_{i,j}^{(5)}, & i = \alpha + 1,\ j = \alpha; \\ \mathrm{Q}_{i,j}^{(6)}, & i \in (\alpha + 1, 2\alpha],\ j = i - 1 - \alpha; \\ \mathrm{Q}_{i,j}^{(7)}, & i = \alpha + 1,\ j = 2\alpha; \\ \mathrm{Q}_{i,j}^{(8)}, & i \in (\alpha + 1, 2\alpha],\ j = i - 1; \\ 0, & \text{otherwise}; \end{cases} \tag{2.207}$$

where

$$\begin{aligned}
\mathrm{Q}_{i,j}^{(4)} &= \frac{\partial \dot{v}_i}{\partial v_i} \cdot \frac{\Delta t^3}{3} + \left(\frac{\partial \dot{v}_i}{\partial p_i} + 1 \right) \cdot \frac{\Delta t^2}{2}, \\
\mathrm{Q}_{i,j}^{(5)} &= \frac{\partial \dot{v}_{i-\alpha}}{\partial v_\alpha} \cdot \frac{\Delta t^3}{3} + \frac{\partial \dot{v}_{i-\alpha}}{\partial p_\alpha} \cdot \frac{\Delta t^2}{2}, \\
\mathrm{Q}_{i,j}^{(6)} &= \frac{\partial \dot{v}_{i-\alpha}}{\partial v_{i-\alpha-1}} \cdot \frac{\Delta t^3}{3} + \frac{\partial \dot{v}_{i-\alpha}}{\partial p_{i-\alpha-1}} \cdot \frac{\Delta t^2}{2}, \\
\mathrm{Q}_{i,j}^{(7)} &= \frac{\partial \dot{v}_{i-\alpha}}{\partial v_\alpha} \cdot \frac{\Delta t^2}{3}, \\
\mathrm{Q}_{i,j}^{(8)} &= \frac{\partial \dot{v}_{i-\alpha}}{\partial v_{i-\alpha-1}} \cdot \frac{\Delta t^2}{3}.
\end{aligned} \tag{2.208}$$

References

[1] D. L. Alspach and H. W. Sorenson. "Nonlinear Bayesian estimation using Gaussian sum approximations". In: *IEEE Transactions on Automatic Control* 17.4 (Aug. 1972), pp. 439–448.

[2] B. D. O. Anderson and J. B. Moore. *Optimal Filtering*. Englewood Cliffs, NJ: Prentice-Hall, 1979.

[4] M. S. Arulampalam et al. "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking". In: *IEEE Transactions on Signal Processing* 50.2 (2002), pp. 174–188.

[6] Y. Bar-Shalom, F. Daum, and J. Huang. "The Probabilistic Data Association Filter". In: *IEEE Control Systems* 29.6 (Dec. 2009), pp. 82–100.

[7] Y. Bar-Shalom, P. K. Willett, and X. Tian. *Tracking and Data Fusion: A Handbook of Algorithms*. YBS Publishing, Apr. 2011.

[9] J.-D. Benamou and Y. Brenier. "A computational fluid mechanics solution to the Monge-Kantorovich mass transfer problem". In: *Numerische Mathematik* 84.3 (2000), pp. 375–393.

[10] V. E. Beneš. "Exact finite-dimensional filters for certain diffusions with nonlinear drift". In: *Stochastics* 5.1-2 (1981), pp. 65–92.

[13] M. Betancourt. "A General Metric for Riemannian Manifold Hamiltonian Monte Carlo". In: *Geometric Science of Information: First International Conference, GSI 2013, Paris, France, August 28-30, 2013. Proceedings*. Ed. by F. Nielsen and F. Barbaresco. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 327–334.

[14] P. Bickel, B. Li, and T. Bengtsson. "Sharp failure rates for the bootstrap particle filter in high dimensions". In: *Pushing the Limits of Contemporary Statistics: Contributions in Honor of Jayanta K. Ghosh*. Ed. by B. Clarke and S. Ghosal. Vol. 3. Collections. Beachwood, Ohio, USA: Institute of Mathematical Statistics, 2008, pp. 318–329.

[15] H. A. P. Blom and E. A. Bloem. "Joint Particle Filtering of Multiple Maneuvering Targets From Unassociated Measurements". In: *Journal of Advances in Information Fusion* 1.1 (2006), pp. 15–34.

[16] H. A. P. Blom and E. A. Bloem. "Probabilistic data association avoiding track coalescence". In: *IEEE Transactions on Automatic Control* 45.2 (Feb. 2000), pp. 247–259.

[19] R. Bucy. "Nonlinear filtering theory". In: *IEEE Transactions on Automatic Control* 10.2 (Apr. 1965), pp. 198–198.

[20] R. S. Bucy and P. D. Joseph. *Filtering for Stochastic Processes with Applications to Guidance*. Ed. by A. C. Publishing. 2nd. John Wiley & Sons, 1968.

[22] A. Budhiraja and G. Kallianpur. "Approximations to the solution of the Zakai equation using multiple Wiener and Stratonovich integral expansions". In: *Stochastics and Stochastic Reports* 56.3-4 (1996), pp. 271–315.

[23] A. Budhiraja and G. Kallianpur. "The Feynman-Stratonovich semigroup and Stratonovich integral expansions in nonlinear filtering". English. In: *Applied Mathematics and Optimization* 35.1 (1997), pp. 91–116.

[24] P. Bunch and S. Godsill. "Particle filtering with progressive Gaussian approximations to the optimal importance density". In: *IEEE 5th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), 2013*. 2013, pp. 360–363.

[25] P. Bunch and S. Godsill. "Approximations of the Optimal Importance Density using Gaussian Particle Flow Importance Sampling". In: *Journal of the American Statistical Association* 111.514 (2016), pp. 748–762.

[30] S. Choi et al. "Discussion and application of the homotopy filter". In: *Proc. SPIE. 8050, Signal Processing, Sensor Fusion, and Target Recognition XX*. Vol. 8050. May 2011, pp. 805021–805021-12.

[35] D. Crisan. "Particle Filters in a Continuous Time Framework". In: *IEEE Nonlinear Statistical Signal Processing Workshop, 2006*. Sept. 2006, pp. 73–78.

[36] D. Crisan, P. Del Moral, and T. J. Lyons. "Interacting Particle Systems Approximations of the Kushner-Stratonovitch Equation". English. In: *Advances in Applied Probability* 31.3 (1999), pp. 819–838.

[37] D. Crisan and K. Li. "Generalised particle filters with Gaussian mixtures". In: *Stochastic Processes and their Applications* 125.7 (2015), pp. 2643–2673.

[38] D. Crisan and T. Lyons. "A particle approximation of the solution of the Kushner-Stratonovitch equation". English. In: *Probability Theory and Related Fields* 115.4 (1999), pp. 549–578.

[39] D. Crisan and J. Xiong. "Approximate McKean-Vlasov representations for a class of SPDEs". In: *Stochastics* 82 (2010), pp. 53–68.

[40] B. Dacorogna and J. Moser. "On a partial differential equation involving the Jacobian determinant". In: *Annales de l'I.H.P. Analyse non linéaire* 7.1 (1990), pp. 1–26.

[41] A. S. Dalalyan. "Theoretical guarantees for approximate sampling from smooth and log-concave densities". In: *ArXiv e-prints* (Dec. 2014). arXiv: 1412.7392 [stat.CO].

[42] A. S. Dalalyan and A. B. Tsybakov. "Sparse Regression Learning by Aggregation and Langevin Monte-Carlo". In: *Journal of Computing and System Sciences* 78.5 (Sept. 2012), pp. 1423–1443.

[43] A. S. Dalalyan. "Theoretical guarantees for approximate sampling from smooth and log-concave densities". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* (2016).

[46] F. Daum and J. Huang. "Curse of dimensionality and particle filters". In: *IEEE Aerospace Conference, 2003*. Vol. 4. Mar. 2003, pp. 1979–1993.

[47] F. Daum and J. Huang. "Particle flow for nonlinear filters, Bayesian decisions and transport". In: *FUSION 2013, Proceedings of the 16th International Conference on Information Fusion*. 2013, pp. 1072–1079.

[48] F. Daum and J. Huang. "Generalized particle flow for nonlinear filters". In: *Proc. SPIE 7698, Signal and Data Processing of Small Targets 2010*. Vol. 7698. Orlando, Florida, Apr. 2010, pp. 76980I–76980I-12.

[49] F. Daum and J. Huang. "Nonlinear filters with log-homotopy". In: *Proc. SPIE 6699, Signal and Data Processing of Small Targets 2007*. Vol. 6699. Signal and Data Processing of Small Targets, 2007. San Diego, California, Sept. 2007, pp. 669918–669918-15.

[50] F. Daum and J. Huang. "Nonlinear filters with particle flow". In: *Proc. SPIE 7445, Signal and Data Processing of Small Targets 2009*. Vol. 7445. Aug. 2009, 74450R–74450R-9.

[51] F. Daum and J. Huang. "Particle flow for nonlinear filters with log-homotopy". In: *Proc. SPIE 6969, Signal and Data Processing of Small Targets 2008*. Vol. 6969. Apr. 2008, pp. 696918–696918-12.

[52] F. Daum and J. Huang. "Particle flow with non-zero diffusion for nonlinear filters, Bayesian decisions and transport". In: *Proc. SPIE 8857, Signal and Data Processing of Small Targets 2013*. Vol. 8745. Sept. 2013, 87450P–87450P-13.

[53] F. Daum, J. Huang, and A. Noushin. "Exact particle flow for nonlinear filters". In: *Proc. SPIE 7697, Signal Processing, Sensor Fusion, and Target Recognition XIX*. Vol. 7697. Apr. 2010, pp. 769704–769704-19.

[54] F. E. Daum. "Exact finite-dimensional nonlinear filters". In: *IEEE Transactions on Automatic Control* 31.7 (July 1986), pp. 616–622.

[56] P. Del Moral, A. Doucet, and A. Jasra. "Sequential Monte Carlo samplers". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 68.3 (2006), pp. 411–436.

[62] J. Deutscher, A. Blake, and I. Reid. "Articulated body motion capture by annealed particle filtering". In: *IEEE Conference on Computer Vision and Pattern Recognition, 2000*. Vol. 2. 2000, pp. 126–133.

[63] T. Ding and M. J. Coates. "Implementation of the Daum-Huang exact-flow particle filter". In: *IEEE Statistical Signal Processing Workshop (SSP)*. 2012, pp. 257–260.

[65] A. Doucet and A. M. Johansen. "A Tutorial on Particle Filtering and Smoothing: Fifteen Years Later". In: *The Oxford Handbook of Nonlinear Filtering*. Ed. by D. Crisan and B. Rozovsky. Oxford University Press, 2009.

[67] S. Duane et al. "Hybrid Monte Carlo". In: *Physics Letters B* 195.2 (1987), pp. 216–222.

[70] L. C. Evans and W. Gangbo. "Differential equations methods for the Monge-Kantorovich mass transfer problem". In: *Memoirs of the American Mathematical Society* 137 (1999), p. 653.

[78] C. W. Gardiner. *Stochastic methods: A Handbook for the Natural and Social Sciences*. 2nd. Springer-Verlag, Berlin–Heidelberg–New York–Tokyo, 1985.

[79] A. Gelman and X.-L. Meng. "Simulating normalizing constants: from importance sampling to bridge sampling to path sampling". In: *Statistical Science* 13.2 (May 1998), pp. 163–185.

[80] M. Girolami and B. Calderhead. "Riemann manifold Langevin and Hamiltonian Monte Carlo methods". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 73.2 (2011), pp. 123–214.

[84] R. Graham. "Statistical Theory of Instabilities in Stationary Nonequilibrium Systems with Applications to Lasers and Nonlinear Optics". In: *Springer Tracts in Modern Physics: Ergebnisse der exakten Naturwissenschaftenc; Volume 66*. Ed. by G. Höhler. Berlin, Heidelberg: Springer Berlin Heidelberg, 1973, pp. 1–97.

[86] C. E. Gutiérrez. *The Monge-Ampère Equation*. Progress in Nonlinear Differential Equations and Their Applications. Birkhäuser Boston, 2001.

[87] J. Hagmar et al. "Optimal parameterization of posterior densities using homotopy". In: *FUSION 2011, Proceedings of the 14th International Conference on Information Fusion*. July 2011, pp. 1–8.

[88] U. D. Hanebeck and J. Steinbring. "Progressive Gaussian filtering based on Dirac Mixture approximations". In: *FUSION 2012, Proceedings of the 15th International Conference on Information Fusion*. July 2012, pp. 1697–1704.

[89] U. D. Hanebeck, K. Briechle, and A. Rauh. "Progressive Bayes: a new framework for nonlinear state estimation". In: *Proc. SPIE. 5099, Multisensor, Multisource Information Fusion: Architectures, Algorithms, and Applications 2003*. Vol. 5099. Apr. 2003, pp. 256–267.

[91] J. Heng, A. Doucet, and Y. Pokern. "Gibbs Flow for Approximate Transport with Applications to Bayesian Computation". In: *ArXiv e-prints* (Sept. 2015). arXiv: 1509.08787 [stat.CO].

[92] N. Higham. "Computing the nearest correlation matrix–a problem from finance". In: *IMA Journal of Numerical Analysis* 22 (2002), pp. 329–343.

[100] H.-K. Janssen. "On a Lagrangean for classical field dynamics and renormalization group calculations of dynamical critical properties". English. In: *Zeitschrift für Physik B Condensed Matter* 23.4 (1976), pp. 377–380.

[101] A. H. Jazwinski. *Stochastic Processes and Filtering Theory*. New York: Academic Press, 1970.

[105] R. Jordan and D. Kinderlehrer. "An Extended Variational Principle". In: *Partial Differential Equations and Applications: Collected Papers in Honor of Carlo Pucci*. Ed. by G. Talenti, E. Vesentini, and P. Marcellini. Lecture Notes in Pure and Applied Mathematics. Taylor & Francis, 1996. Chap. 18.

[106] R. Jordan, D. Kinderlehrer, and F. Otto. "The Variational Formulation of the Fokker–Planck Equation". In: *SIAM Journal on Mathematical Analysis* 29.1 (1998), pp. 1–17.

[109] M. Khan and M. Ulmke. "Improvements in the implementation of log-homotopy based particle flow filters". In: *FUSION 2015, Proceedings of the 18th International Conference on Information Fusion*. July 2015, pp. 74–81.

[110] M. Klaas, N. de Freitas, and A. Doucet. "Toward Practical N^2 Monte Carlo: the Marginal Particle Filter". In: *ArXiv e-prints* (July 2012). arXiv: 1207.1396 [stat.CO].

[111] J. H. Kotecha and P. M. Djurić. "Gaussian particle filtering". In: *IEEE Transactions on Signal Processing* 51 (2003), pp. 2592–2601.

[114] H. J. Kushner. "On the differential equations satisfied by conditional probability densities of Markov processes, with applications". In: *Journal of the SIAM, Series A (Control)* 2.1 (1964), pp. 106–119.

[116] H. Lamb. *Hydrodynamics*. University Press, 1895.

[119] D. A. Levin, Y. Peres, and E. L. Wilmer. "Markov chains and mixing times". In: With a chapter on coupling from the past by James G. Propp and David B. Wilson. Providence, R.I. American Mathematical Society, 2009. Chap. 4.

[122] X. R. Li and Z. Zhao. "Measuring Estimator's Credibility: Noncredibility Index". In: *FUSION 2006, Proceedings of the 9th International Conference on Information Fusion*. July 2006, pp. 1–8.

[133] A. Marrs, S. Maskell, and Y. Bar-Shalom. "Expected likelihood for tracking in clutter with particle filters". In: *Proc. SPIE. 4728, Signal and Data Processing of Small Targets 2002*. Vol. 4728. Aug. 2002, pp. 230–239.

[137] S. Meyn and R. L. Tweedie. *Markov Chains and Stochastic Stability*. 2nd. New York, NY, USA: Cambridge University Press, 2009.

[140] G. N. Milstein, J. G. M. Schoenmakers, and V. Spokoiny. "Transition Density Estimation for Stochastic Differential Equations via Forward-Reverse Representations". English. In: *Bernoulli* 10.2 (2004), pp. 281–312.

[142] J. Moser. "On the Volume Elements on a Manifold". In: *Transactions of the American Mathematical Society* 120.2 (1965), pp. 286–294.

[147] R. M. Neal. "Annealed importance sampling". English. In: *Statistics and Computing* 11.2 (2001), pp. 125–139.

[148] R. M. Neal. *Bayesian Learning for Neural Networks*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 1996.

[149] E. Novak. "Some Results on the Complexity of Numerical Integration". In: *Monte Carlo and Quasi-Monte Carlo Methods: MCQMC, Leuven, Belgium, April 2014*. Ed. by R. Cools and D. Nuyens. Springer International Publishing, 2016, pp. 161–183.

[151] N. Oudjane and C. Musso. "Progressive correction for regularized particle filters". In: *FUSION 2000, Proceedings of the 3rd International Conference on Information Fusion*. Vol. 2. July 2000, THB2/10–THB2/17.

[152] T. Ozaki. "A bridge between nonlinear time series models and nonlinear stochastic dynamical systems: a local linearization approach". In: *Statistica Sinica* 2.1 (1992), pp. 113–135.

[153] L. Y. Pao. "Multisensor multitarget mixture reduction algorithms for tracking". In: *Journal of Guidance, Control, and Dynamics* 17.6 (Nov. 1994), pp. 1205–1211.

[155] N. S. Pillai, A. M. Stuart, and A. H. Thiéry. "Optimal Scaling and Diffusion Limits for the Langevin Algorithm in High Dimensions". In: *Annals of Applied Probability* 22.6 (Dec. 2012), pp. 2320–2356.

[159] R. C. Rao. "Information and the accuracy attainable in the estimation of statistical parameters". In: *Bulletin of Calcutta Mathematical Society* 37 (1945), pp. 81–91.

[160] P. Rebeschini and R. van Handel. "Can local particle filters beat the curse of dimensionality?" In: *Annals of Applied Probability* 25.5 (Oct. 2015), pp. 2809–2866.

[161] S. Reich. "A dynamical systems framework for intermittent data assimilation". English. In: *BIT Numerical Mathematics* 51.1 (2011), pp. 235–249.

[162] S. Reich. "A Guided Sequential Monte Carlo Method for the Assimilation of Data into Stochastic Dynamical Systems". English. In: *Recent Trends in Dynamical Systems*. Ed. by A. Johann et al. Vol. 35. Springer Proceedings in Mathematics & Statistics. Springer Basel, 2013, pp. 205–220.

[165] H. Risken. *The Fokker-Planck Equation: Methods of Solution and Applications*. 2nd, Springer-Verlag, 1989.

[167] G. O. Roberts, A. Gelman, and W. R. Gilks. "Weak convergence and optimal scaling of random walk Metropolis algorithms". In: *Annals of Applied Probability* 7.1 (Feb. 1997), pp. 110–120.

[168] G. O. Roberts and O. Stramer. "Langevin Diffusions and Metropolis-Hastings Algorithms". English. In: *Methodology And Computing In Applied Probability* 4.4 (2002), pp. 337–357.

[169] G. O. Roberts and J. S. Rosenthal. "Optimal Scaling of Discrete Approximations to Langevin Diffusions". In: *Journal of the Royal Statistical Society. Series B (Statistical Methodology)* 60.1 (1998), pp. 255–268.

[172] S. Sarkar et al. "A Kushner-Stratonovich Monte Carlo filter applied to nonlinear dynamical system identification". In: *Physica D: Nonlinear Phenomena* 270 (2014), pp. 46–59.

[175] F. Septier and G. W. Peters. "Langevin and Hamiltonian Based Sequential MCMC for Efficient Bayesian Filtering in High-Dimensional Spaces". In: *IEEE Journal of Selected Topics in Signal Processing* 10.2 (Mar. 2016), pp. 312–327.

[177] B. W. Silverman. *Density Estimation for Statistics and Data Analysis*. London: Chapman and Hall, Apr. 1986.

[183] C. Snyder et al. "Obstacles to High-Dimensional Particle Filtering". In: *Monthly Weather Review* 136.12 (Dec. 2008), pp. 4629–4640.

[185] H. W. Sorenson and D. L. Alspach. "Recursive Bayesian Estimation Using Gaussian Sums". In: *Automatica* 7.4 (July 1971), pp. 465–479.

[187] R. L. Stratonovich. "Optimum nonlinear systems which bring about a separation of a signal with constant parameters from noise". In: *Radiofizika* 2.6 (1959), pp. 892–901.

[191] G. Terejanu et al. "A novel Gaussian Sum Filter Method for accurate solution to the nonlinear filtering problem". In: *FUSION 2008, Proceedings of the 11th International Conference on Information Fusion*. June 2008, pp. 1–8.

[193] M. Treiber and A. Kesting. *Traffic flow dynamics: Data, Models and Simulation*. Springer-Verlag Berlin Heidelberg, 2013.

[197] M. Venugopal, R. M. Vasu, and D. Roy. "An Ensemble Kushner-Stratonovich-Poisson Filter for Recursive Estimation in Nonlinear Dynamical Systems". In: *IEEE Transactions on Automatic Control* 61.3 (Mar. 2016), pp. 823–828.

[198] C. Vergé et al. "On parallel implementation of sequential Monte Carlo methods: the island particle model". English. In: *Statistics and Computing* 25.2 (2015), pp. 243–260.

[199] C. Villani. *Topics in Optimal Transportation*. Graduate studies in Mathematics. American Mathematical Society, 2003.

[211] M. Zakai. "On the optimal filtering of diffusion processes". English. In: *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete* 11.3 (1969), pp. 230–243.

3

Multi-target Tracking of Dependent Targets

As discussed in Chapter 0, a typical assumption of multi-target trackers considers that target states are mutually independent conditioned on the observations. Although this assumption simplifies the multi-target state estimation, since in that case the uncertainty of each target state can be fully represented independently of the others, it results in relevant inaccuracies when tracked objects get into proximity and stay in this condition for a considerable duration of time. Estimating the correlations between objects in mutual proximity can be achieved by filters that estimate the object states jointly (as a single multi-target state), along with their covariances, but often constitute a computationally feasible solution only for a small number of targets. The challenge of representing target states and their mutual dependencies in a concise, accurate, and scalable way is treated in this chapter.

In this context, we develop a scalable multi-target tracking framework that considers dependency between target states due to association uncertainty and enables track management. Our method builds on a target tracker previously developed to efficiently maintain target dependency structures, incorporating mechanisms for initiating new tracks via birth process and by inferring existence probabilities to cater for track confirmation and deletion. The method can also be seen as an extension of the Joint Integrated Probabilistic Data Association (JIPDA) to include target state dependencies and birth processes. The proposed framework does not maintain a joint probability distribution over all the target states, which is infeasible except when the number of targets is small, but rather relies on target dependency trees on which discrete probability distributions of adjacent targets are calculated. These dependency trees constitute probabilistic graphical models, where each node represents a target-related discrete random variable and the edges represent the probabilistic relations between nodes. Each node variable can take different values, one for each mixture component of the corresponding target state, with probabilities given by a discrete marginal distribution, and each edge is described by a conditional probability distribution relating two nodes. These probabilistic trees are predicted and updated at each time step such that the information lost in the approximation is minimal. For computing the marginal association probabilities, an exact, very efficient algorithm known as Efficient Hypothesis Management (EHM) is adopted in its most general form, which enables computations for pairs of dependent targets. A numerical experiment shows that the proposed filter reduces the incidence of track swapping and substantially improves tracking capability when information sensible for association disambiguation are occasionally available.

3.1 Overview

Multi-target tracking can be regarded as the subdomain of Signal Processing dedicated to estimating states of several objects from a sequence of observations corrupted by noise, under detection uncertainty and the possibility of false alarms. Pioneer methods for multi-target tracking have resorted to multivariate Bayesian statistics for describing targets' states and have proposed enumeration-based probabilistic models of the measurement-to-target association uncertainty to enable custody of tracked objects in their estimates. No wonder these descriptions result in a combinatorial association problem with exponential complexity on the number of targets (and of measurements). Established representatives of such methods include the celebrated Multi-Hypothesis Tracking (MHT) [163] and Joint Probabilistic Data Association (JPDA) [73]. These filters have become traditional due to their easy incorporation of well-known filtering techniques (e.g., Kalman measurement update), accurate approximations of posterior multi-target state densities and an intuitive formalism.

On the other hand, a different family of filters approaches multi-target Bayesian inference in a different way, where data association is avoided by means of a permutational joint state description while promoting a succinct mathematical formulation. In this formulation, a collection of target states is considered as a random set-valued state and the collection of sensor measurements is treated as a random set-valued observation. Often, the cost of avoiding the measurement-to-target association is having estimates that do not keep track identities explicitly. Celebrated representatives of this class are the Probabilistic Hypothesis Density (PHD) filter [126], and the Multi-target Multi-Bernoulli filter [131]. Recent advances worth mentioning include the Labeled Multi-Bernoulli filter (Labeled MeMBer) [164] and the Distinguishable and Independent Stochastic Populations (DISP) filter [58]. The Labeled Multi-Bernoulli filter generalizes random finite sets to incorporate object labels so that target identities can be estimated, however, this approach reintroduces the combinatorial complexity when producing estimates. The DISP filter relies on the classical formalism of stochastic population processes [143], along with a measure-theoretical description for treating uncertainties about populations in different levels of abstraction. The DISP incurs in a computational complexity similar to that of the MHT.

In the traditional target tracking formulation, filters maintain a marginal distribution on each target state and approximate the joint distribution as a product of the marginals, i.e., they assume that the targets are independent. While this constitutes a good approximation for many practical cases, it becomes coarse when targets get close to one another for several time steps. One possible solution to consider dependencies between targets is estimating a complete joint distribution over the states of all targets, such as in the coupled JPDA approaches [7, 16, 15], however, this solution is only feasible for a few targets owing to the high computational effort involved in updating high-dimensional probability distributions. In contrast, we propose that the multi-target joint distribution be represented via a set of disjoint trees over the set of targets, and so is approximated by using the pairwise distributions of adjacent targets in the same tree. The target trees are constructed and updated such that they preserve the maximum available information, in the same ethos as that of Chow-Liu trees [31]. In our proposed algorithm, the computational requirements should grow subexponentially on the number of targets.

In this chapter, we adopt the traditional approach akin to the Joint Integrated Probabilistic

Data Association (JIPDA) [144], but introducing a dependency structure between target states and relying on a highly efficient algorithm for computing marginal association probabilities (EHM) [135, 95]. Our method is an extension of the filter proposed in [94], incorporating mechanisms for initiating new tracks via birth process, inferring existence probabilities to cater for track confirmation and deletion, and generalizing the procedure for constructing and updating target trees. The reasons for adopting the traditional multi-target tracking formulation are simple: i) filters based on random finite sets and stochastic populations do not offer a clear way to describe dependency between targets to this day, ii) the complexity of the JIPDA framework when solved via the EHM is scalable to a moderately high number of targets, iii) probabilistic Bayesian structures are well established and can be directly incorporated in the traditional formulation (see, for example, [207, 208, 117]).

Based on these facts, our approach focuses on the problem of tracking multiple targets from measurements where the correspondence of measurements to targets is unknown and the targets may depend on one another. The intent is developing a new filter to reduce track swapping, i.e. when target identities and their associated tracks get swapped. This situation is particularly prominent when targets move into mutual proximity, which renders the filter unable to distinguish them, and then they move apart such that the filter may be induced to estimate the wrong modes of the multi-target distribution. With exception of the MHT and its variations, traditional filters are compelled to make a "hard decision" when close targets get apart, which can go wrong with substantial probability, and once this decision has gone wrong these filters cannot revise their estimates later on when disambiguating information might become available. In the literature, this problem is referred to as *track coalescence*, and most filters proposed to tackle it also arrive at a dilemma where a hard decision must be made, usually in the form of an optimization problem, e.g., producing a maximum-a-posteriori estimate [16] or maximizing information [190, 117]. The point we wish to make is that upon maintaining target dependency, the decision is softer in the sense that a wrong estimate does not permanently affect future estimates. This is expected because the dependency structure keeps "memory" of the association ambiguities, which enables the filter to revise its estimates later on. In this view, the tracker proposed herein defers a deal of its decision ability to future steps when disambiguating information becomes available.

The chapter is organized as follows. In Section 3.2 we state the multi-target tracking problem, present usual assumptions, and extend the assumptions to formalize our filter. In Section 3.3 we derive the prediction step, and in Section 3.4 we derive the measurement update step in a general context. Details on the computation of posterior marginal association probabilities are discussed in Section 3.5, including a detailed description of EHM and presenting the inference equations for EHM 3. Section 3.6 is dedicated to explaining how target trees are constructed and updated, according to new association hypotheses. Section 3.7 describes the mixture reduction method used in our filter and the criteria for track management. Section 3.8 presents a numerical experiment that shows the capabilities of our method. Section 3.9 concludes.

3.2 Problem statement and assumptions

As it is customary in stochastic filtering problems, let us fix an underlying probability space $(\Omega, \mathcal{F}(\mathcal{X}), \mathcal{P})$ on which random experiments will be defined, where Ω is the sample space, $\mathcal{F}(\mathcal{X})$

is a σ-algebra on some (state) space $\mathcal{X}$, characterizing a collection of probabilistic events, and $\mathcal{P}$ is a probability measure that associates probabilities to events. It is generally assumed that each target follows a state process $\{\mathrm{x}_t\}_{t\geq 0}$, $\mathrm{x}_t \in \mathcal{X} \subseteq \mathbb{R}^{d_x}$, from which realizations can be evaluated at subsequent discrete-time steps $t \in \{t_0, t_1, \ldots, t_k, \ldots | k \in \mathbb{N}_0\}$, and so we denote $\mathrm{x}_k \coloneqq \mathrm{x}_{t_k}$ hereafter. Likewise, an observation process, $\{\mathrm{y}_t\}_{t\geq 0}$, $\mathrm{y}_t \in \mathcal{Y} \subseteq \mathbb{R}^{d_y}$, is defined to describe measurements about target states, from which discrete-time outcomes, $\mathrm{y}_k \coloneqq \mathrm{y}_{t_k}$, are provided by at least one sensor. The description also establishes the measurable spaces of the state and observation processes respectively, $(\mathcal{X}, \mathcal{F}_t(\mathcal{X}))$ and $(\mathcal{Y}, \mathcal{F}_t(\mathcal{Y}))$, with σ-fields $\mathcal{F}_t(\mathcal{X}) \triangleq \sigma\{\mathrm{x}_t | t \geq 0\}$ and $\mathcal{F}_t(\mathcal{Y}) \triangleq \sigma\{\mathrm{y}_t | t > 0\}$ that compose the filtrations $\{\mathcal{F}_t(\mathcal{X})\}_{t\geq 0}$ and $\{\mathcal{F}_t(\mathcal{Y})\}_{t\geq 0}$ of the σ-algebra $\mathcal{F}$. From now on, when talking about the state and observation processes, we refer to their discrete-time versions, $\{\mathrm{x}_k\}_{k\in\mathbb{N}_0}$ and $\{\mathrm{y}_k\}_{k\in\mathbb{N}}$.

In this chapter we index objects by finite sets of distinct natural numbers as

$$\mathcal{I}_n \coloneqq \{i_1, \ldots, i_n | i_1, \ldots, i_n \in \mathbb{N},\, i_1 \neq \cdots \neq i_n\}$$

for n elements, and any such index set will be of the same form. In multiple target tracking, a number n of targets may possibly be in the scene. By indexing all targets in the scene by $\mathcal{I}_{n,k} \coloneqq \{i_1, \ldots, i_n\}$, each target state is described as $\mathrm{x}_k^{(i)}$ such that $i \in \mathcal{I}_{n,k}$. The set of all targets in the scene at time step k will be denoted as $\mathrm{x}_k^{(1:n)} = \mathrm{x}_k^{\mathcal{I}_{n,k}} \coloneqq \{\mathrm{x}_k^{(i)} | i \in \mathcal{I}_{n,k}\}$. In favor of simplicity, throughout this chapter the notation $\mathrm{x}_k^{(1:n)}$ is used interchangeably with $\mathrm{x}_k^{\mathcal{I}_{n,k}}$ to represent the finite set containing all objects $\mathrm{x}_k^{(i)}$ indexed by $\mathcal{I}_{n,k}$. Though we acknowledge that an index set $\mathcal{I}_{n,k}$ might contain natural numbers (identifiers) not necessarily in the interval $[1..n]$, in which case $\mathrm{x}_k^{(1:n)}$ would be an abuse of notation. Similarly, a number m of measurements about targets are collected, indexed by $\mathcal{I}_{m,k} \coloneqq \{j_1, \ldots, j_m\}$, and so each measurement can be identified as $\mathrm{y}_k^{(j)}$ such that $j \in \mathcal{I}_{m,k}$. The complete set of measurements is denoted as $\mathrm{Y}_k \triangleq \mathrm{y}_k^{(1:m)} \coloneqq \{\mathrm{y}_k^{(j)} | j \in \mathcal{I}_{m,k}\}$ and we write the history of observations up to time step k as $\mathrm{Y}_{1:k} \coloneqq \{\mathrm{Y}_1, \ldots, \mathrm{Y}_k\}$. Multi-object state and observation spaces can be denoted as the Cartesian products $\mathcal{X}^n = \prod_{i\in\mathcal{I}_{n,k}} \mathcal{X}$ and $\bar{\mathcal{Y}}^m = \prod_{j\in\mathcal{I}_{m,k}} \bar{\mathcal{Y}}$ respectively, where $\bar{\mathcal{Y}} \triangleq \mathcal{Y} \cup \mathcal{Y}_{\mathrm{fa}}$ and, in this context, the observation process is assumed to admit false alarms $\mathrm{y}_{k,\mathrm{fa}}^{(j)} \in \mathcal{Y}_{\mathrm{fa}}$ as realizations and missed detections may be present ($m \leq n$).

The ultimate goal of any multiple-target tracker is to infer estimates about target states conditioned on the information provided by the observation history, that is

$$\mathcal{P}_k[\varphi] = \mathbb{E}\left[\varphi(\mathrm{x}_k^{(1:n)}) | \mathcal{F}_k(\bar{\mathcal{Y}}^m)\right] \equiv \int_{\mathcal{X}^n} \varphi(\mathrm{x}_k^{(1:n)}) \mathcal{P}_k(d\mathrm{x}_k^{(1:n)}, \mathrm{Y}_{1:k}), \tag{3.1}$$

for a test function $\varphi : \mathcal{X}^n \to \mathcal{X}'$, measurable in $\mathcal{F}_k(\mathcal{X}^n) = \sigma\{\mathrm{x}_t^{(1:n)} | 0 < t \leq t_k\}$, where $\mathcal{P}_k(\cdot, \mathrm{Y}_{1:k})$ is a posterior multi-target probability measure, and $\mathrm{x}_k^{(1:n)}$ should be understood in terms of the joint event $\mathrm{X}_k^{(1:n)} = \bigcap_{i=1}^n \mathrm{X}_k^{(i)}$ in the measurable space $(\mathcal{X}^n, \mathcal{F}_k(\mathcal{X}^n))$. In practical terms, computing estimates as in (3.1) involves expressing the posterior multi-target measure explicitly, which constitutes the main scope of almost all multiple target trackers. In order to express the multi-target posterior measure, trackers require further assumptions, which are typically given as follows.

Assumption 3.1. *Each target moves independently of one another, with motion modeled by a single-target Markov transition density $p_t(\mathrm{x}_k^{(i)} | \mathrm{x}_{k-1}^{(i)})$. Each target state is assumed to follow a*

state equation of the form

$$\mathrm{x}_k^{(i)} = \mathrm{f}(\mathrm{x}_{k-1}^{(i)}) + \mathrm{g}(\xi_k),\ i \in \mathcal{I}_{n,k-1}, \tag{3.2}$$

where $\mathrm{f} : \mathcal{X} \to \mathcal{X}$ *is the state transition function, and* $\mathrm{g} : \mathcal{X} \to \mathcal{X}$ *is a function of the state noise process* $\{\xi_k\}_{k\in\mathbb{N}_0}$, $\xi_k \in \mathcal{X}$, *with independent realizations over time steps.*

Assumption 3.2. *Measurements generated from targets are independent on one another, with single-target likelihood function* $\ell_j(\mathrm{x}_k) = p_\ell(\mathrm{y}_k^{(j)}|\mathrm{x}_k)$. *Each target state is assumed to produce at most one measurement, obtained according to an observation equation of the form*

$$\mathrm{y}_k^{(i)} = \mathrm{h}(\mathrm{x}_k^{(i)}) + \upsilon_k,\ i \in \mathcal{I}_{n,k}, \tag{3.3}$$

where $\mathrm{h} : \mathcal{X} \to \mathcal{Y}$ *is the observation function, and* $\{\upsilon_k\}_{k\in\mathbb{N}_0}$, $\upsilon_k \in \mathcal{Y}$, *is the observation noise process, whose realizations are independent over distinct measurement outcomes, and independent of the state noise process* $\{\xi_k\}_{k\in\mathbb{N}_0}$.

Assumption 3.3. *Each sensor detects each target with state* $\mathrm{x}_k^{(i)}$ *with probability* $p_d(\mathrm{x}_k^{(i)})$. *Hence, there may be misdetections with probability* $q_d(\mathrm{x}_k^{(i)}) = 1 - p_d(\mathrm{x}_k^{(i)})$.

Assumption 3.4. *False alarms (clutter) may affect the observation process, being independent of the target-originated measurements. The number of false alarms is assumed Poisson-distributed, i.e.,* $m_{\mathrm{fa}} \sim Pois(m|\lambda_{\mathrm{fa}})$, *where* λ_{fa} *is expected number of false alarms per time frame, and the position of each false alarm is assumed uniformly distributed in a surveillance region occupying a volume* V, *i.e.* $\mathrm{y}_{\mathrm{fa}} \sim \mathcal{U}(\partial V)$ *where* ∂V *is the boundary (surface) of the surveillance region. We define the expected volumetric density of false alarms per time frame according to* $\lambda_{V,\mathrm{fa}} = \lambda_{\mathrm{fa}}/V$. *At each time step* k, *the complete set of measurements is then given by*

$$\mathrm{y}_k^{\mathcal{I}_{m,k}} = \mathrm{y}_k^{\mathcal{I}_{n,k}} \cup \mathrm{y}_{\mathrm{fa}}^{\mathcal{I}_{m_{\mathrm{fa}},k}}$$

and sensors cannot distinguish target-generated measurements from false alarms.

In the Joint Probabilistic Data Association (JPDA) framework, estimating the joint posterior distribution $\mathcal{P}_k$ is structured by the additional assumption that target's states are independent conditioned on the observation information. Denoting p a probability density of $\mathcal{P}_k$ with respect to the Lebesgue measure, the posterior multi-target density under the JPDA framework would be of the product form, $p(\mathrm{x}_k^{(1:n)}|\mathrm{Y}_{1:k}) = \prod_{i=1}^{n} p(\mathrm{x}_k^{(i)}|\mathrm{Y}_{1:k})$, where $p(\mathrm{x}_k^{(i)}|\mathrm{Y}_{1:k})$ are the marginal probability densities of each target state. This form simplifies the multi-target density representation since it can be completely described by the individual target marginal densities. However, this simplicity comes at a cost: loss of ability to explain dependency between targets that may probabilistically share the same measurement in the observation path.

The uncertainty of whether a measurement may have originated from each of two (or more) targets arises when they are in sufficient proximity such that the measurement is likely to have come from either of them. In addition, such uncertainty is exacerbated if targets remain in mutual proximity for several time steps and, in that case, assuming target state independence would discard sensible information. With the intent of partially avoiding this information loss, Horridge & Maskell [94] proposed a filter where target dependency is maintained via a probabilistic graph structure, in the form of a tree, that expresses pairwise relations across

targets. In this context, the filter proposed by Horridge & Maskell [94] makes two additional assumptions as follows.

Assumption 3.5. *The marginal probability densities of each target state assume a mixture form according to*

$$p(\mathrm{x}_k^{(i)}|\mathrm{Y}_{1:k}) = \sum_{c_k^{(i)} \in \Gamma_k^{(i)}} p(\mathrm{x}_k^{(i)}|c_k^{(i)}, \mathrm{Y}_{1:k}) p(c_k^{(i)}|\mathrm{Y}_{1:k}), \tag{3.4}$$

where $c_k^{(i)}$ *is a discrete random variable taking values in a discrete countable set*

$$\Gamma_k^{(i)} = \{\gamma_1^{(i)}, \gamma_2^{(i)}, \ldots, \gamma_{n_{\gamma,k}(i)}^{(i)}\},$$

with posterior probabilities expressed as $p(c_k^{(i)}|\mathrm{Y}_{1:k}) = \Pr\{c_k^{(i)}|\mathrm{Y}_{1:k}\}$*, and where* $n_{\gamma,k}(i) = |\Gamma_k^{(i)}|$ *is the event set cardinality for the component* $c_k^{(i)}$*. The target states are assumed independent conditioned on the set of component variables* $c_k^{(1:n)} = \{c_k^{(1)}, \ldots, c_k^{(n)}\}$*, i.e.*

$$p(\mathrm{x}_k^{(1:n)}|c_k^{(1:n)}, \mathrm{Y}_{1:k}) = \prod_{i=1}^{n} p(\mathrm{x}_k^{(i)}|c_k^{(1:n)}, \mathrm{Y}_{1:k}),$$

but the components themselves may be dependent on each other over all targets.

Assumption 3.6. *The target dependency structure is described by a probabilistic tree,* $G_T = (\mathcal{V}_T, \mathcal{E}_T)$*, where* $\mathcal{V}_T = \mathcal{I}_{n,k}$ *is the set of vertices and* $\mathcal{E}_T = \{(u,v)|u,v \in \mathcal{V}_T, u \neq v\}$ *is a set of vertice pairs that explains edges in the tree. In this tree, each node represents a target* i*, probabilistically described by a (discrete) random variable,* $c_k^{(i)}$*, and the edges represent probabilistic dependency between two targets. Adjacent nodes in the tree are related by discrete conditional distributions of the form* $p(c_k^{(i)}|c_k^{\mathrm{pa}(i)}, \mathrm{Y}_{1:k})$*, where* $\mathrm{pa}(i)$ *stands for the parent of node* i*. In addition,*

$$i \perp\!\!\!\perp \mathrm{an}(i) \setminus \mathrm{pa}(i) \,|\, \mathrm{pa}(i),\ i \in \mathcal{I}_{n,k}, \tag{3.5}$$

where $\mathrm{an}(i)$ *stands for the set of all ancestors of node* i*, which means that* $p(c_k^{(i)}|c_k^{\mathrm{an}(i)}, \mathrm{Y}_{1:k}) = p(c_k^{(i)}|c_k^{\mathrm{pa}(i)}, \mathrm{Y}_{1:k})$*. The dependency structure over all targets can be expressed by the joint distribution:*

$$p(c_k^{(1:n)}|\mathrm{Y}_{1:k}) = \prod_{i=1}^{n} p(c_k^{(i)}|c_k^{\mathrm{pa}(i)}, \mathrm{Y}_{1:k}). \tag{3.6}$$

We will increment those assumptions to: (i) model the uncertainty of target existence, (ii) maintain a set of unconfirmed tracks, (iii) model the possibility that new targets may appear in the scene. These assumptions will be used to develop a mechanism for track management, involving initiating, confirming, and deleting tracks. As a consequence, conditional probability densities of target states in *Assumption 3.5* will also be conditioned on the target existence. This is particularly useful for deciding whether a track (on a hypothetical target) should be confirmed or discarded, based on its evaluated probability of existence.

Assumption 3.7. *Targets can exist or not, in a probabilistic sense, according to a binary existence random variable* $e_k \in \{0,1\}$ *at time step* k*. The existence process,* $\{e_k\}_{k\in\mathbb{N}}$*, is assumed to follow a first-order Markov process. Let* $E_k^{(i)} \triangleq \{e_k^{(i)} = 1\}$ *be the event that target* i *exists, and* $\bar{E}_k^{(i)} \triangleq \{e_k^{(i)} = 0\}$ *the complementary event. The posterior existence probability of each target is described by* $p(E_k^{(i)}|\mathrm{Y}_{1:k})$*. The Markov transition kernel of the existence process, denoted by*

$p(e_k^{(i)}|e_{k-1}^{(i)})$, *models possibilities that an existing target at time step* $k-1$ *may die (disappear) or survive until the next time step. The probability of survival is denoted as* $p_s = p(E_k^{(i)}|E_{k-1}^{(i)})$. *Marginal probability densities of each target state will be conditioned on target existence and read as*

$$p(\mathrm{x}_k^{(i)}|E_k^{(i)}, \mathrm{Y}_{1:k}) = \sum_{c_k^{(i)} \in \Gamma_k^{(i)}} p(\mathrm{x}_k^{(i)}|c_k^{(i)}, E_k^{(i)}, \mathrm{Y}_{1:k}) p(c_k^{(i)}|\mathrm{Y}_{1:k}). \tag{3.7}$$

Assumption 3.8. *Targets that have never been detected, or have been detected but show probability of existence within the interval* $p(E_k^{(i)}|\mathrm{Y}_{1:k}) \in (\tau_{\mathrm{del}}, \tau_{\mathrm{conf}})$, *where* τ_{del} *is a lower threshold for track deletion and* τ_{conf} *is an upper threshold for track confirmation, take part in set of* $n_{u,k}$ *unconfirmed targets, indexed by* $\mathcal{I}_{u,k} = \{i_1^u, \ldots, i_{n_{u,k}}^u\}$. *For simplicity, each unconfirmed target has a single state component, distributed according to* $p(\mathrm{x}_k^{(i)}|E_k^{(i)}, \mathrm{Y}_{1:k})$. *While keeping track of unconfirmed targets, if* $p(E_k^{(i)}|\mathrm{Y}_{1:k}) > \tau_{\mathrm{conf}}$ *for a target* $i \in \mathcal{I}_{u,k}$, *then the corresponding track is confirmed, removed from the unconfirmed set, and incorporated in a set of* $n_{c,k}$ *confirmed targets, indexed by* $\mathcal{I}_{c,k} = \{i_1^c, \ldots, i_{n_{c,k}}^c\}$. *After each filtering cycle, if any confirmed or unconfirmed target has* $p(E_k^{(i)}|\mathrm{Y}_{1:k}) < \tau_{\mathrm{del}}$, *then its track is deleted.*

Assumption 3.9. *New targets may appear independently of the existing targets at time step* k, *being indexed by* $\mathcal{I}_{b,k} = \{i_1^b, \ldots, i_{n_{b,k}}^b\}$. *The number* $n_b = n_{b,k}$ *of new appearing targets is assumed to be Poisson-distributed according to* $n_b \sim Pois(n|\lambda_b)$ *where* $\lambda_b = \lambda_{V,b} V$ *is expected number of new targets per time frame, and* $\lambda_{V,b}$ *is the expected volumetric density of new targets per time frame. Each new target* $i \in \mathcal{I}_{b,k}$ *is assumed to be spatially distributed according to* $p_{\mathrm{birth}}(\mathrm{x}_k^{(i)})$. *Each new target has existence probability given by* $p_b \triangleq \lambda_b$.

The filter that results from *Assumptions 3.1–3.9* requires a solution for the data association problem. Later on we will explain how uncertainty of measurement-to-target associations will be taken into account, but for now it suffices to say that the posterior (marginal) association probabilities will be exactly computed by an algorithm reminiscent of Pearl's algorithm [154] in a Bayesian network[1], called Efficient Hypothesis Management (EHM) [135, 95, 94]. We will adopt the third variation of EHM as in [94], which handles multiple state (mixture) components per target. The resulting filter will be referred to as JPDA-EHM3 hereafter.

3.3 Prediction step

The prediction step for the JPDA-EHM3 is supported by *Assumptions 3.1, 3.5, 3.7, 3.8 and 3.9.* As per *Assumptions 3.8 and 3.9*, the problem is partitioned into the sets of confirmed targets, indexed by $\mathcal{I}_{c,k-1} = \{i_1^c, \ldots, i_{n_{c,k-1}}^c\}$, unconfirmed targets indexed by $\mathcal{I}_{u,k-1} = \{i_1^u, \ldots, i_{n_{u,k-1}}^u\}$, and newborn targets indexed by $\mathcal{I}_{b,k} = \{i_1^b, \ldots, i_{n_{b,k}}^b\}$. Given that $\mathcal{I}_{c,k|k-1} \coloneqq \mathcal{I}_{c,k-1}$, note that

[1] Which can also be seen as belief propagation.

for confirmed targets:

$$
\begin{aligned}
&p(\mathrm{x}_k^{(i)}|E_k^{(i)}, \mathrm{Y}_{1:k-1}) \\
&\quad = \int_{\mathcal{X}} p_t(\mathrm{x}_k^{(i)}|\mathrm{x}_{k-1}^{(i)}) \left(\sum_{c_{k-1}^{(i)} \in \Gamma_{k-1}^{(i)}} p(\mathrm{x}_{k-1}^{(i)}|c_{k-1}^{(i)}, E_k^{(i)}, \mathrm{Y}_{1:k-1}) p(c_{k-1}^{(i)}|\mathrm{Y}_{1:k-1}) \right) d\mathrm{x}_{k-1}^{(i)}, \\
&\quad \equiv \sum_{c_{k-1}^{(i)} \in \Gamma_{k-1}^{(i)}} p(\mathrm{x}_k^{(i)}|c_{k-1}^{(i)}, E_k^{(i)}, \mathrm{Y}_{1:k-1}) p(c_{k-1}^{(i)}|\mathrm{Y}_{1:k-1}),\ i \in \mathcal{I}_{c,k|k-1}, \quad (3.8)
\end{aligned}
$$

$$
\begin{aligned}
p(E_k^{(i)}|\mathrm{Y}_{1:k-1}) &= \sum_{e_k^{(i)} \in \{0,1\}} p(E_k^{(i)}|e_{k-1}^{(i)}) p(e_{k-1}^{(i)}|\mathrm{Y}_{1:k-1}) \\
&= p(E_k^{(i)}|E_{k-1}^{(i)}) p(E_{k-1}^{(i)}|\mathrm{Y}_{1:k-1}) \\
&= p_s \cdot p(E_{k-1}^{(i)}|\mathrm{Y}_{1:k-1}),\ i \in \mathcal{I}_{c,k|k-1}, \quad (3.9)
\end{aligned}
$$

where

$$
\begin{aligned}
p(\mathrm{x}_k^{(i)}|c_{k-1}^{(i)}, E_k^{(i)}, \mathrm{Y}_{1:k-1}) &= p(\mathrm{x}_k^{(i)}|c_{k-1}^{(i)}, E_{k-1}^{(i)}, \mathrm{Y}_{1:k-1}), \\
p(\mathrm{x}_k^{(i)}|E_k^{(i)}, \mathrm{Y}_{1:k-1}) &= p(\mathrm{x}_k^{(i)}|E_k^{(i)}, E_{k-1}^{(i)}, \mathrm{Y}_{1:k-1}),
\end{aligned}
$$

because confirmed targets that existed at time step $k-1$ are predicted as continuing to exist at time step k with probability $p(E_k^{(i)}|\mathrm{Y}_{1:k-1})$. Now, for unconfirmed targets in $\mathcal{I}_{u,k|k-1} \coloneqq \mathcal{I}_{u,k-1} \cup \mathcal{I}_{b,k}$, observe that

$$
\begin{aligned}
p(\mathrm{x}_k^{(i)}|E_k^{(i)}, \mathrm{Y}_{1:k-1}) &\equiv p(\mathrm{x}_k^{(i)}|E_k^{(i)}, e_{k-1}^{(i)}, \mathrm{Y}_{1:k-1}) \\
&= \int_{\mathcal{X}} p_t(\mathrm{x}_k^{(i)}|\mathrm{x}_{k-1}^{(i)}) p(\mathrm{x}_{k-1}^{(i)}|E_k^{(i)}, e_{k-1}^{(i)}, \mathrm{Y}_{1:k-1}) d\mathrm{x}_{k-1}^{(i)} \\
&= \begin{cases} p(\mathrm{x}_k^{(i)}|E_k^{(i)}, E_{k-1}^{(i)}, \mathrm{Y}_{1:k-1}) = p(\mathrm{x}_k^{(i)}|E_{k-1}^{(i)}, \mathrm{Y}_{1:k-1}), & e_{k-1}^{(i)} = 1,\ i \in \mathcal{I}_{u,k-1}, \\ p(\mathrm{x}_k^{(i)}|E_k^{(i)}, \bar{E}_{k-1}^{(i)}, \mathrm{Y}_{1:k-1}) = p_{\mathrm{birth}}(\mathrm{x}_k^{(i)}), & e_{k-1}^{(i)} = 0,\ i \in \mathcal{I}_{b,k}, \end{cases} \quad (3.10)
\end{aligned}
$$

$$
p(E_k^{(i)}|\mathrm{Y}_{1:k-1}) = \begin{cases} p_s \cdot p(E_{k-1}^{(i)}|\mathrm{Y}_{1:k-1}), & e_{k-1}^{(i)} = 1,\ i \in \mathcal{I}_{u,k-1}, \\ p_b, & e_{k-1}^{(i)} = 0,\ i \in \mathcal{I}_{b,k}, \end{cases} \quad (3.11)
$$

where p_b is the probability that a new target i will appear in the scene with probability density given by $p_{\mathrm{birth}}(\mathrm{x}_k^{(i)})$. Note that possibly existing targets at time step $k-1$ are predicted as continuing to exist with probability $p(E_k^{(i)}|\mathrm{Y}_{1:k-1})$. In addition, in (3.9) and (3.11), $p(E_k^{(i)}|\bar{E}_{k-1}^{(i)}) = 0$ for possibly existing targets at time step $k-1$, and $p(\bar{E}_{k-1}^{(i)}|\mathrm{Y}_{1:k-1}) = 1$ for targets born at time step k.

3.4 Measurement update step

The measurement update step for the JPDA-EHM3 relies on the *Assumptions 3.2–3.9*, and will be performed over the entire set of targets, indexed by $\mathcal{I}_{n,k} = \mathcal{I}_{u,k} \cup \mathcal{I}_{c,k}$. Note that the method described in this section is applicable to both confirmed and unconfirmed targets,

because unconfirmed target densities could be interpreted as mixtures with a single component.

For simplicity of notation, we write $n = n_k^c + n_k^u$ and $m = m_k$ in this section. Let $\{a_k^{(i)}\}_{k\in\mathbb{N}}$ be the association process, described by a discrete random variable that associates a target in $\mathcal{I}_n = \{i_1, \ldots, i_n\}$ to a single measurement in $\mathcal{I}_{m,0} \coloneqq \mathcal{I}_m \cup \{0\}$, i.e. $a_k^{(i)} : \mathcal{I}_n \to \mathcal{I}_m \cup \{0\}$, where $a_k^{(i)} = 0$ denotes the event in which the ith target generates no measurement. Because each target generates at most one measurement per *Assumption 3.2*, the nonzero values of $a_k^{(1:n)} = \{a_k^{(1)}, \ldots, a_k^{(n)}\}$ must be distinct. In this context we define the set of all possible joint associations over exactly $|\mathcal{S}|$ targets, for some subset $\mathcal{S} \subseteq \mathcal{I}_n$, such that nonzero elements are distinct, as

$$\mathcal{A}_{\mathcal{S},k} \triangleq \{a_k^{\mathcal{S}} | a_k^{(\ell)} \in \mathcal{I}_{m,0}, \forall \ell \in \mathcal{S},\, a_k^{(i)} \neq a_k^{(j)} \neq 0 \implies i \neq j\}. \tag{3.12}$$

Clearly, the joint posterior multi-target density can be described as the mixture

$$\begin{aligned} &p(\mathrm{x}_k^{(1:n)} | E_k^{(1:n)}, \mathrm{Y}_{1:k}) \\ &\qquad = \sum_{c_k^{(1:n)}} \sum_{a_k^{(1:n)}} \prod_{i=1}^{n} p(\mathrm{x}_k^{(i)} | c_{k-1}^{(i)}, a_k^{(i)}, E_k^{(i)}, \mathrm{Y}_{1:k}) p(c_{k-1}^{(1:n)}, a_k^{(1:n)} | E_k^{(1:n)}, \mathrm{Y}_{1:k}), \end{aligned} \tag{3.13}$$

where $p(\mathrm{x}_k^{(i)} | c_{k-1}^{(i)}, a_k^{(i)}, E_k^{(i)}, \mathrm{Y}_{1:k})$ is a single-target posterior density in the usual sense (e.g., via Kalman filter) and where only joint associations permitted by *Assumption 3.2* are considered in the joint event space, that is,

$$p(c_{k-1}^{(1:n)}, a_k^{(1:n)} | E_k^{(1:n)}, \mathrm{Y}_{1:k}) \coloneqq p(c_{k-1}^{(1:n)}, a_k^{(1:n)} | a_k^{(1:n)} \in \mathcal{A}_{\mathcal{I}_n,k}, E_k^{(1:n)}, \mathrm{Y}_{1:k}). \tag{3.14}$$

By applying Bayes' rule one can write

$$\begin{aligned} &p(c_{k-1}^{(1:n)}, a_k^{(1:n)}, E_k^{(1:n)} | \mathrm{Y}_{1:k}) \\ &\qquad = \frac{p(\mathrm{y}_k^{(1:m)} | c_{k-1}^{(1:n)}, a_k^{(1:n)}, E_k^{(1:n)}, m) \cdot p(a_k^{(1:n)}, E_k^{(1:n)}, m | \mathrm{Y}_{1:k-1}) \cdot p(c_{k-1}^{(1:n)} | \mathrm{Y}_{1:k-1})}{p(\mathrm{y}_k^{(1:m)} | \mathrm{Y}_{1:k-1})}. \end{aligned} \tag{3.15}$$

Let a binary random variable $d_k^{(i)} \in \{0, 1\}$ describe the ith target detection, where $D_k^{(i)} \triangleq \{d_k^{(i)} = 1\}$ is the detection event and $\bar{D}_k^{(i)} \triangleq \{d_k^{(i)} = 0\}$ its complementary event, set the total number of detections as $m_d \coloneqq \sum_{i=1}^{n} d_k^{(i)}$, and let $\mathcal{I}_{d,k} \coloneqq \{i \in \mathcal{I}_{n,k} | d_k^{(i)} = 1\}$ and $\mathcal{I}_{\bar{d},k} = \mathcal{I}_{n,k} \setminus \mathcal{I}_{d,k}$ be the index sets of detected targets and undetected targets, respectively, in a given joint association event $a_k^{(1:n)}$. Also, denote the joint detection as $d_k^{(1:n)} = \{d_k^{(1)}, \ldots, d_k^{(n)}\}$ and consider random permutations over the set of measurement indexes as $\sigma_m : \mathcal{I}_m \to \mathcal{I}_m$, each equally probable with $p(\sigma_m | m) = 1/m!$. As observed in [95], statistics on a joint event $(a_k^{(1:n)}, m)$ can be inferred from a joint event $(d_k^{(1:n)}, \sigma_m)$ according to

$$p(a_k^{(1:n)}, m | \mathrm{Y}_{1:k-1}) = (m - m_d)! \cdot p(d_k^{(1:n)}, \sigma_m | \mathrm{Y}_{1:k-1}), \tag{3.16}$$

that is, knowing the prior probability of a joint detection configuration with a specific measurement permutation allows calculating the prior probability of joint association with a certain number of measurements. The term $(m - m_d)!$ accounts for all ways of ordering non-detections in $a_k^{(1:n)}$ since permuting missed detections produces the same effect in the prior association distribution. The joint prior association probability $p(a_k^{(1:n)}, E_k^{(1:n)}, m | \mathrm{Y}_{1:k-1})$ must be handled

carefully since in joint events where $a_k^{(i)} = 0$, the associations must also account for the fact that i may not exist, i.e., a missed detection can happen either if a target does exist or not. Denote the event that detected targets exist as $E_k^{\mathcal{I}_{d,k}} = \{E_k^{(i)}, \forall i \in \mathcal{I}_{n,k} | d_k^{(i)} = 1\}$, and clearly $(d_k^{(1:n)}, E_k^{\mathcal{I}_{d,k}}) \equiv d_k^{(1:n)}$ (a detected target always exists), then we can write

$$\begin{aligned} p(a_k^{(1:n)}, E_k^{(1:n)}, m|\mathrm{Y}_{1:k-1}) &= (m-m_d)! \cdot p(d_k^{(1:n)}, E_k^{(1:n)}, \sigma_m, |\mathrm{Y}_{1:k-1}) \\ &= (m-m_d)! \cdot p(\sigma_m|m) \cdot p(m|d_k^{(1:n)}) \cdot p(d_k^{(1:n)}, E_k^{(1:n)}|\mathrm{Y}_{1:k-1}) \\ &= (m-m_d)! \cdot p(\sigma_m|m) \cdot p(m|d_k^{(1:n)}) \cdot \alpha^{\mathcal{I}_{\bar{d},k}} p(d_k^{(1:n)}, E_k^{\mathcal{I}_{d,k}}|\mathrm{Y}_{1:k-1}), \\ &= (m-m_d)! \cdot p(\sigma_m|m) \cdot p(m|d_k^{(1:n)}) \cdot \alpha^{\mathcal{I}_{\bar{d},k}} p(d_k^{(1:n)}|\mathrm{Y}_{1:k-1}), \quad (3.17) \end{aligned}$$

where the prior probability of joint detections equals

$$\begin{aligned} &p(d_k^{(1:n)}|\mathrm{Y}_{1:k-1}) \\ &\quad \equiv p(d_k^{(1:n)}, E_k^{\mathcal{I}_{d,k}}|\mathrm{Y}_{1:k-1}) = \prod_{i=1}^{n} p(\bar{D}_k^{(i)}|\mathrm{Y}_{1:k-1})^{1-d_k^{(i)}} p(D_k^{(i)}, E_k^{(i)}|\mathrm{Y}_{1:k-1})^{d_k^{(i)}} \\ &= \prod_{i=1}^{n} \left(\sum_{e_k^{(i)}} p(\bar{D}_k^{(i)}|e_k^{(i)}, \mathrm{Y}_{1:k-1}) p(e_k^{(i)}|\mathrm{Y}_{1:k-1}) \right)^{1-d_k^{(i)}} \left(p(D_k^{(i)}|E_k^{(i)}, \mathrm{Y}_{1:k-1}) p(E_k^{(i)}|\mathrm{Y}_{1:k-1}) \right)^{d_k^{(i)}} \\ &= \textstyle\prod_{i=1}^{n} \left(1 - p(D_k^{(i)}|E_k^{(i)}, \mathrm{Y}_{1:k-1}) p(E_k^{(i)}|\mathrm{Y}_{1:k-1})\right)^{1-d_k^{(i)}} \left(p(D_k^{(i)}|E_k^{(i)}, \mathrm{Y}_{1:k-1}) p(E_k^{(i)}|\mathrm{Y}_{1:k-1})\right)^{d_k^{(i)}}, \quad (3.18) \end{aligned}$$

and where the following factor is introduced

$$\begin{aligned} \alpha^{\mathcal{I}_{\bar{d},k}} &\triangleq \prod_{i \in \mathcal{I}_{\bar{d},k}} \frac{p(\bar{D}_k^{(i)}, E_k^{(i)}|\mathrm{Y}_{1:k-1})}{p(\bar{D}_k^{(i)}|\mathrm{Y}_{1:k-1})} = p(E_k^{\mathcal{I}_{\bar{d},k}}|d_k^{(1:n)}, E_k^{\mathcal{I}_{d,k}}, \mathrm{Y}_{1:k-1}) \\ &\equiv p(E_k^{(1:n)}|d_k^{(1:n)}) \equiv p(E_k^{(1:n)}|a_k^{(1:n)}). \quad (3.19) \end{aligned}$$

Observe that, indeed,

$$p(d_k^{(1:n)}, E_k^{(1:n)}|\mathrm{Y}_{1:k-1}) = \alpha^{\mathcal{I}_{\bar{d},k}} p(d_k^{(1:n)}, E_k^{\mathcal{I}_{d,k}}|\mathrm{Y}_{1:k-1}) \equiv p(E_k^{(1:n)}|d_k^{(1:n)}) p(d_k^{(1:n)}|\mathrm{Y}_{1:k-1}), \quad (3.20)$$

and so

$$\begin{aligned} &p(a_k^{(1:n)}, E_k^{(1:n)}, m|\mathrm{Y}_{1:k-1}) \\ &\qquad = \alpha^{\mathcal{I}_{\bar{d},k}} p(a_k^{(1:n)}, E_k^{\mathcal{I}_{d,k}}, m|\mathrm{Y}_{1:k-1}) \equiv p(E_k^{(1:n)}|a_k^{(1:n)}) p(a_k^{(1:n)}, m|\mathrm{Y}_{1:k-1}). \quad (3.21) \end{aligned}$$

The following expressions are applicable in (3.18):

$$p(\bar{E}_k^{(i)}|\mathrm{Y}_{1:k-1}) = 1 - p(E_k^{(i)}|\mathrm{Y}_{1:k-1}), \tag{3.22}$$

$$p(D_k^{(i)}|E_k^{(i)}, \mathrm{Y}_{1:k-1}) \triangleq \int_{\mathcal{X}} p_d(\mathrm{x}_k^{(i)}) p(\mathrm{x}_k^{(i)}|E_k^{(i)}, \mathrm{Y}_{1:k-1}) d\mathrm{x}_k^{(i)}, \tag{3.23}$$

$$p(\bar{D}_k^{(i)}|E_k^{(i)}, \mathrm{Y}_{1:k-1}) = 1 - p(D_k^{(i)}|E_k^{(i)}, \mathrm{Y}_{1:k-1}), \tag{3.24}$$

$$p(\bar{D}_k^{(i)}|\bar{E}_k^{(i)}, \mathrm{Y}_{1:k-1}) = 1, \tag{3.25}$$

$$p(D_k^{(i)}|\bar{E}_k^{(i)}, \mathrm{Y}_{1:k-1}) = 1 - p(\bar{D}_k^{(i)}|\bar{E}_k^{(i)}, \mathrm{Y}_{1:k-1}) = 0. \tag{3.26}$$

Ultimately, computing marginal association probabilities in the JPDA framework shall be based on

$$\begin{aligned} p(c_{k-1}^{(1:n)}, a_k^{(1:n)}, E_k^{\mathcal{I}_{d,k}}|\mathrm{Y}_{1:k}) &\equiv p(c_{k-1}^{(1:n)}, a_k^{(1:n)}|\mathrm{Y}_{1:k}) \\ &= \frac{p(\mathrm{y}_k^{(1:m)}|c_{k-1}^{(1:n)}, a_k^{(1:n)}, E_k^{\mathcal{I}_{d,k}}, m) \cdot p(a_k^{(1:n)}, E_k^{\mathcal{I}_{d,k}}, m|\mathrm{Y}_{1:k-1}) \cdot p(c_{k-1}^{(1:n)}|\mathrm{Y}_{1:k-1})}{p(\mathrm{y}_k^{(1:m)}|\mathrm{Y}_{1:k-1})}, \end{aligned} \tag{3.27}$$

so that

$$\begin{aligned} p(c_{k-1}^{(1:n)}, a_k^{(1:n)}, E_k^{(1:n)}|\mathrm{Y}_{1:k}) &= p(E_k^{(1:n)}|a_k^{(1:n)}) p(c_{k-1}^{(1:n)}, a_k^{(1:n)}|\mathrm{Y}_{1:k}) \\ &\equiv \alpha^{\mathcal{I}_{\bar{d},k}} p(c_{k-1}^{(1:n)}, a_k^{(1:n)}, E_k^{\mathcal{I}_{d,k}}|\mathrm{Y}_{1:k}). \end{aligned} \tag{3.28}$$

The approach formulated in [95] can be generalized to give

$$\begin{aligned} p(\mathrm{y}_k^{(1:m)}|c_{k-1}^{(1:n)}, a_k^{(1:n)}, m) &\equiv p(\mathrm{y}_k^{(1:m)}|c_{k-1}^{(1:n)}, a_k^{(1:n)}, E_k^{\mathcal{I}_{d,k}}, m) \\ &= \int_{\mathcal{X}^n} p(\mathrm{y}_k^{(1:m)}|c_{k-1}^{(1:n)}, a_k^{(1:n)}, E_k^{\mathcal{I}_{d,k}}, \mathrm{x}_k^{(1:n)}, m) p(\mathrm{x}_k^{(1:n)}|c_{k-1}^{(1:n)}, a_k^{(1:n)}, E_k^{\mathcal{I}_{d,k}}, \mathrm{Y}_{1:k-1}) d\mathrm{x}_k^{(1:n)} \\ &= \mathbb{1}_{\mathcal{A}_{\mathcal{I}_n,k}}(a_k^{(1:n)}) \prod_{j=1}^{m-m_d} \mathcal{U}(\partial V) \cdot \prod_{i=1}^{n} \int_{\mathcal{X}} \ell_{a_k^{(i)}}(\mathrm{x}_k^{(i)}) p(\mathrm{x}_k^{(i)}|c_{k-1}^{(i)}, E_k^{(i)}, \mathrm{Y}_{1:k-1}) d\mathrm{x}_k^{(i)} \\ &= \mathbb{1}_{\mathcal{A}_{\mathcal{I}_n,k}}(a_k^{(1:n)}) V^{-(m-m_d)} \cdot \prod_{i=1}^{n} \int_{\mathcal{X}} \ell_{a_k^{(i)}}(\mathrm{x}_k^{(i)}) p(\mathrm{x}_k^{(i)}|c_{k-1}^{(i)}, E_k^{(i)}, \mathrm{Y}_{1:k-1}) d\mathrm{x}_k^{(i)}, \end{aligned} \tag{3.29}$$

where

$$\mathbb{1}_{\mathcal{A}_{\mathcal{I}_n,k}}(a_k^{(1:n)}) = \begin{cases} 1, & a_k^{(1:n)} \in \mathcal{A}_{\mathcal{I}_n,k}, \\ 0, & \text{otherwise}, \end{cases}$$

is the indicator function, and we note that each of the $m - m_d$ measurements corresponding to non-detections are false alarms spatially distributed according to $\mathcal{U}(\partial V) = V^{-1}$. As a

consequence of (3.17), (3.18) and (3.21):

$$\begin{aligned}
&p(a_k^{(1:n)}, E_k^{\mathcal{I}_{d,k}}, m|\mathrm{Y}_{1:k-1}) \equiv p(a_k^{(1:n)}, m|\mathrm{Y}_{1:k-1}) \\
&= (m-m_d)! \cdot p(d_k^{(1:n)}, E_k^{\mathcal{I}_{d,k}}, \sigma_m|\mathrm{Y}_{1:k-1}) \\
&= (m-m_d)! \cdot p(\sigma_m|m) \cdot p(m|d_k^{(1:n)}) \cdot p(d_k^{(1:n)}, E_k^{\mathcal{I}_{d,k}}|\mathrm{Y}_{1:k-1}) \\
&= (m-m_d)! \cdot \frac{1}{m!} \cdot \mathrm{Pois}(m-m_d|\lambda_{\mathrm{fa}}) \cdot p(d_k^{(1:n)}|\mathrm{Y}_{1:k-1}) \\
&= \frac{e^{-\lambda_{\mathrm{fa}}}(\lambda_{V,\mathrm{fa}}V)^{m-m_d}}{m!} \prod_{i=1}^{n} \Big(1 - p(D_k^{(i)}|E_k^{(i)}, \mathrm{Y}_{1:k-1}) p(E_k^{(i)}|\mathrm{Y}_{1:k-1})\Big)^{1-d_k^{(i)}} \\
&\times \Big(p(D_k^{(i)}|E_k^{(i)}, \mathrm{Y}_{1:k-1}) p(E_k^{(i)}|\mathrm{Y}_{1:k-1})\Big)^{d_k^{(i)}},
\end{aligned} \tag{3.30}$$

where a detection can only happen if a target exists, and a missed detection can happen either due to the target non-existence or due to a sensor misdetection. Substituting (3.29) and (3.30) in (3.27) we obtain

$$\begin{aligned}
&p(c_{k-1}^{(1:n)}, a_k^{(1:n)}, E_k^{\mathcal{I}_{d,k}}|\mathrm{Y}_{1:k}) \\
&\quad = \frac{e^{-\lambda_{\mathrm{fa}}}\lambda_{V,\mathrm{fa}}^{m-n}}{m! p(\mathrm{y}_k^{(1:m)}|\mathrm{Y}_{1:k-1})} \mathbb{1}_{\mathcal{A}_{\mathcal{I}_n,k}}(a_k^{(1:n)}) \cdot p(c_{k-1}^{(1:n)}|\mathrm{Y}_{1:k-1}) \prod_{i=1}^{n} \psi(a_k^{(i)}, E_k^{(i)}|c_{k-1}^{(i)}),
\end{aligned} \tag{3.31}$$

where

$$\begin{aligned}
&\psi(a_k^{(i)}, E_k^{(i)}|c_{k-1}^{(i)}) \\
&\quad = \begin{cases}
\lambda_{V,\mathrm{fa}} \left(1 - p(E_k^{(i)}|\mathrm{Y}_{1:k-1}) \int_{\mathcal{X}} p_d(\mathrm{x}_k^{(i)}) p(\mathrm{x}_k^{(i)}|E_k^{(i)}, \mathrm{Y}_{1:k-1}) d\mathrm{x}_k^{(i)}\right), & a_k^{(i)} = 0, \\
p(E_k^{(i)}|\mathrm{Y}_{1:k-1}) \int_{\mathcal{X}} p_d(\mathrm{x}_k^{(i)}) \ell_{a_k^{(i)}}(\mathrm{x}_k^{(i)}) p(\mathrm{x}_k^{(i)}|c_{k-1}^{(i)}, E_k^{(i)}, \mathrm{Y}_{1:k-1}) d\mathrm{x}_k^{(i)}, & a_k^{(i)} \neq 0,
\end{cases}
\end{aligned} \tag{3.32}$$

and where we use the notation $\ell_{a_k^{(i)}}(\mathrm{x}_k^{(i)}) = p_\ell(\mathrm{y}_k^{a_k^{(i)}}|\mathrm{x}_k^{(i)})$ for the likelihood function. Using (3.6) from *Assumption 3.6* in (3.31):

$$\begin{aligned}
p(c_{k-1}^{(1:n)}, a_k^{(1:n)}, E_k^{\mathcal{I}_{d,k}}|\mathrm{Y}_{1:k}) &\propto \mathbb{1}_{\mathcal{A}_{\mathcal{I}_n,k}}(a_k^{(1:n)}) \cdot \prod_{i=1}^{n} p(c_{k-1}^{(i)}|c_{k-1}^{\mathrm{pa}(i)}, \mathrm{Y}_{1:k-1}) \prod_{j=1}^{n} \psi(a_k^{(j)}, E_k^{(j)}|c_{k-1}^{(j)}) \\
&\propto \prod_{i=1}^{n} \mathbb{1}_{\breve{\mathcal{A}}_{i,k}}(a_k^{(i),\mathrm{an}(i)}) \psi(a_k^{(i)}, E_k^{(i)}|c_{k-1}^{(i)}) p(c_{k-1}^{(i)}|c_{k-1}^{\mathrm{pa}(i)}, \mathrm{Y}_{1:k-1}),
\end{aligned} \tag{3.33}$$

where we define a succinct notation of the indicator function, on a per target basis, as

$$\mathbb{1}_{\breve{\mathcal{A}}_{i,k}}(a_k^{(i),\mathrm{an}(i)}) \triangleq \mathbb{1}_{\mathcal{A}_{\mathrm{an}(i)\cup\{i\},k}}(a_k^{(i),\mathrm{an}(i)}), \tag{3.34}$$

with $a_k^{(i),\mathrm{an}(i)} \coloneqq \{a_k^{(i)}\} \cup \{a_k^{\mathrm{an}(i)}\}$. The indicator function in (3.33) guarantees the validity of associations including the ith target and the set of all its ancestors $\mathrm{an}(i)$. The measurement-

update problem is solved by finding the pairwise associating marginals

$$\begin{aligned} p(c_{k-1}^{(i),\mathrm{pa}(i)}, a_k^{(i),\mathrm{pa}(i)} | E_k^{(i),\mathrm{pa}(i)}, \mathrm{Y}_{1:k}) &= \frac{p(c_{k-1}^{(i)}, c_{k-1}^{\mathrm{pa}(i)}, a_k^{(i)}, a_k^{\mathrm{pa}(i)}, E_k^{(i)}, E_k^{\mathrm{pa}(i)} | \mathrm{Y}_{1:k})}{p(E_k^{(i)}, E_k^{\mathrm{pa}(i)} | \mathrm{Y}_{1:k})} \\ &= p(E_k^{(i),\mathrm{pa}(i)} | a_k^{(i),\mathrm{pa}(i)}) \frac{p(c_{k-1}^{(i),\mathrm{pa}(i)}, a_k^{(i),\mathrm{pa}(i)} | \mathrm{Y}_{1:k})}{p(E_k^{(i),\mathrm{pa}(i)} | \mathrm{Y}_{1:k})} \\ &= \alpha^{\mathcal{I}_{\bar{d},k} \cap \{i,\mathrm{pa}(i)\}} \frac{p(c_{k-1}^{(i),\mathrm{pa}(i)}, a_k^{(i),\mathrm{pa}(i)} | \mathrm{Y}_{1:k})}{p(E_k^{(i),\mathrm{pa}(i)} | \mathrm{Y}_{1:k})}. \end{aligned} \tag{3.35}$$

which requires application of a sum-product algorithm to (3.33) analogous to belief propagation, and considering the factor

$$\alpha^{\mathcal{I}_{\bar{d},k} \cap \{i,\mathrm{pa}(i)\}} = \prod_{\ell \in \mathcal{I}_{\bar{d},k} \cap \{i,\mathrm{pa}(i)\}} \frac{p(\bar{D}_k^{(\ell)}, E_k^{(\ell)} | \mathrm{Y}_{1:k-1})}{p(\bar{D}_k^{(\ell)} | \mathrm{Y}_{1:k-1})}. \tag{3.36}$$

As we will explain later on, a näive implementation of the sum-product algorithm is problematic because it involves complexity exponential in the number of targets. Instead, we shall resort to an exact algorithm called EHM, sub-exponential in number of targets, as proposed in [95, 94]. The pairwise associating marginals (3.35) enable representation of pairwise posterior densities according to

$$\begin{aligned} &p(\mathrm{x}_k^{(i),\mathrm{pa}(i)} | E_k^{(i),\mathrm{pa}(i)}, \mathrm{Y}_{1:k}) \\ &= \textstyle\sum_{c_{k-1}^{(i),\mathrm{pa}(i)}} \sum_{a_{k-1}^{(i),\mathrm{pa}(i)}} p(c_{k-1}^{(i),\mathrm{pa}(i)}, a_k^{(i),\mathrm{pa}(i)} | E_k^{(i),\mathrm{pa}(i)}, \mathrm{Y}_{1:k}) \prod_{\ell \in \{i,\mathrm{pa}(i)\}} p(\mathrm{x}_k^{(\ell)} | c_{k-1}^{(\ell)}, a_k^{(\ell)}, E_k^{(\ell)}, \mathrm{Y}_{1:k}), \end{aligned} \tag{3.37}$$

from which marginal densities for each target state can be obtained as

$$p(\mathrm{x}_k^{(i)} | E_k^{(i)}, \mathrm{Y}_{1:k}) = \int_{\mathcal{X}} p(\mathrm{x}_k^{(i),\mathrm{pa}(i)} | E_k^{(i),\mathrm{pa}(i)}, \mathrm{Y}_{1:k}) d\mathrm{x}_k^{\mathrm{pa}(i)}. \tag{3.38}$$

The posterior existence probability is computed by

$$\begin{aligned} p(E_k^{(i)} | \mathrm{Y}_{1:k}) &= \sum_{a_k^{(i)}} p(a_k^{(i)}, E_k^{(i)} | \mathrm{Y}_{1:k}) = \sum_{a_k^{(i)}} p(E_k^{(i)} | a_k^{(i)}) p(a_k^{(i)} | \mathrm{Y}_{1:k}) \\ &= \sum_{a_k^{(i)}} \alpha^{\mathcal{I}_{\bar{d},k} \cap \{i\}} p(a_k^{(i)}, E_k^{\mathcal{I}_{d,k} \cap \{i\}} | \mathrm{Y}_{1:k}) = \alpha^{(i)} p(a_k^{(i)} = 0 | \mathrm{Y}_{1:k}) + \sum_{a_k^{(i)} \neq 0} \alpha^{\emptyset} p(a_k^{(i)} | \mathrm{Y}_{1:k}) \\ &= \frac{p(\bar{D}_k^{(i)}, E_k^{(i)} | \mathrm{Y}_{1:k-1})}{p(\bar{D}_k^{(i)} | \mathrm{Y}_{1:k-1})} p(a_k^{(i)} = 0 | \mathrm{Y}_{1:k}) + \sum_{a_k^{(i)} \neq 0} p(a_k^{(i)}, E_k^{(i)} | \mathrm{Y}_{1:k}) \\ &= \frac{(1 - p(D_k^{(i)} | E_k^{(i)}, \mathrm{Y}_{1:k-1})) p(E_k^{(i)} | \mathrm{Y}_{1:k-1})}{1 - p(D_k^{(i)} | E_k^{(i)}, \mathrm{Y}_{1:k-1}) p(E_k^{(i)} | \mathrm{Y}_{1:k-1})} p(a_k^{(i)} = 0 | \mathrm{Y}_{1:k}) + \sum_{a_k^{(i)} \neq 0} p(a_k^{(i)} | \mathrm{Y}_{1:k}), \end{aligned} \tag{3.39}$$

where $p(a_k^{(i)} | \mathrm{Y}_{1:k}) \equiv p(a_k^{(i)}, E_k^{(i)} | \mathrm{Y}_{1:k})$ for any $a_k^{(i)} \neq 0$ and

$$p(a_k^{(i)}|\mathrm{Y}_{1:k}) = \sum_{c_{k-1}^{(1:n)}} \sum_{a_k^{(1:n)} \setminus a_k^{(i)} | a_k^{(1:n)} \in \mathcal{A}_{\mathcal{I}_n,k}} p(c_{k-1}^{(1:n)}, a_k^{(1:n)}, E_k^{\mathcal{I}_{d,k}}|\mathrm{Y}_{1:k}). \tag{3.40}$$

The formulation derived in this section is akin to the Joint Integrated Probabilistic Data Association (JIPDA) filter [144], but incorporating dependent mixture components across targets. Our derivation is similar to the procedure proposed in [29], but introducing the factor $\alpha^{\mathcal{I}_{\bar{d},k}}$ in the joint space.

3.5 Computing marginal probability of component associations

In this section, we shall assume that all targets take part in single target tree, G_T, in view of *Assumption 3.6*, since different trees would imply independent groups of targets that could be treated separately, i.e., one tree per cluster of dependent targets. The following notation is used to describe the dependency structure in G_T:

- i represents the node $i \in \mathcal{I}_n = \{i_1, \ldots, i_n\}$ in G_T, corresponding to the ith target,
- $\mathrm{pa}(i)$ is the parent of a node i,
- $\mathrm{ch}(i)$ is the set of all children of node i,
- $\mathrm{an}(i)$ is the set of all ancestors of node i,
- $\mathrm{de}(i)$ is the set of all descendants of node i,
- $\mathrm{sb}(i)$ is the set of all siblings of node i, that is, $\mathrm{sb}(i) \triangleq \mathrm{ch}(\mathrm{pa}(i)) \setminus \{i\}$,
- $\overline{\mathrm{de}}(i)$ is the set of all nodes that are not i or any of its descendants, that is, $\overline{\mathrm{de}}(i) \triangleq \mathcal{I}_n \setminus (\{i\} \cup \mathrm{de}(i))$,
- $\mathrm{cs}(i)$ is the set of cousins of node i, composed of all non-descendants nodes that are not ancestors of i, i.e. $\mathrm{cs}(i) \triangleq \overline{\mathrm{de}}(i) \setminus \mathrm{an}(i)$.

As touched on before, the ultimate goal of the JPDA-EHM3 tracker is estimating marginal posterior densities of target states, either considering pairwise dependencies as in (3.37) or expressing single-target marginal densities as in (3.38). Computing these densities require marginal association probabilities $p(c_{k-1}^{(i),\mathrm{pa}(i)}, a_k^{(i),\mathrm{pa}(i)}|E_k^{(i),\mathrm{pa}(i)}, \mathrm{Y}_{1:k})$ for target-pair state densities, or $p(c_{k-1}^{(i)}, a_k^{(i)}|E_k^{(i)}, \mathrm{Y}_{1:k})$ for single-target state densities. For simplicity of notation we omit the conditioning on $E_k^{(i)}$ in this section. Note that a näive computation of marginal association

probabilities would involve marginalizing (3.33) directly as

$$p(c_{k-1}^{(i)}, a_k^{(i)}|\mathrm{Y}_{1:k}) \propto \sum_{a_k^{\overline{\mathrm{de}}(i)}} \sum_{c_k^{\overline{\mathrm{de}}(i)}} p(c_{k-1}^{\overline{\mathrm{de}}(i)}, a_k^{\overline{\mathrm{de}}(i)}|\mathrm{Y}_{1:k}) \cdot \mathbb{1}_{\breve{\mathcal{A}}_{i,k}}(a_k^{(i),\mathrm{an}(i)}) \psi(a_k^{(i)}|c_{k-1}^{(i)}) p(c_{k-1}^{(i)}|c_{k-1}^{\mathrm{pa}(i)}, \mathrm{Y}_{1:k-1}) \times \left(\sum_{a_k^{\mathrm{de}(i)}} \sum_{c_k^{\mathrm{de}(i)}} p(c_{k-1}^{\mathrm{de}(i)}, a_k^{\mathrm{de}(i)}|c_{k-1}^{(i)}, \mathrm{Y}_{1:k}) \right), \quad (3.41)$$

where

$$p(c_{k-1}^{\overline{\mathrm{de}}(i)}, a_k^{\overline{\mathrm{de}}(i)}|\mathrm{Y}_{1:k}) \propto \prod_{\ell \in \overline{\mathrm{de}}(i)} \mathbb{1}_{\breve{\mathcal{A}}_{\ell,k}}(a_k^{(\ell),\mathrm{an}(\ell)}) \psi(a_k^{(\ell)}|c_{k-1}^{(\ell)}) p(c_{k-1}^{(\ell)}|c_{k-1}^{\mathrm{pa}(\ell)}, \mathrm{Y}_{1:k-1}), \quad (3.42)$$

$$p(c_{k-1}^{\mathrm{de}(i)}, a_k^{\mathrm{de}(i)}|c_{k-1}^{(i)}, \mathrm{Y}_{1:k}) \propto \prod_{\ell \in \mathrm{de}(i)} \mathbb{1}_{\breve{\mathcal{A}}_{\ell,k}}(a_k^{(\ell),\mathrm{an}(\ell)}) \psi(a_k^{(\ell)}|c_{k-1}^{(\ell)}) p(c_{k-1}^{(\ell)}|c_{k-1}^{\mathrm{pa}(\ell)}, \mathrm{Y}_{1:k-1}). \quad (3.43)$$

Although (3.41) could be rearranged to be computed recursively via Pearl's algorithm [154], it is evident that the computational effort would grow exponentially in the number of targets. We work around this computational problem by adopting the algorithm called Efficient Hypothesis Management (EHM 3).

3.5.1 The EHM network

The fundamental idea of the Efficient Hypothesis Management algorithm is to simplify the structure of association hypotheses by eliminating redundancies. Hence, the EHM applies an efficient inference algorithm (Pearl's algorithm) over the association structure to compute marginal association probabilities. We illustrate the concept by forming the association hypotheses for the example given in Table 3.1, which gives the tree of hypotheses depicted in Figure 3.1. In this tree of hypotheses, nodes represent targets and each edge represents a measurement assignment to the child node target. The reader should not confuse the tree of hypotheses with the target tree stated in *Assumption 3.6*: the tree of hypotheses used herein is just a graphic way of illustrating how enumeration of association events would be geometrically complex in the classical approach.

Target	Measurements to associate
T1	M0, M1, M2, M3
T2	M0, M2, M3
T3	M0, M3, M4
T4	M0, M4

Table 3.1: Association example

Redundancies arise in the hypotheses enumeration structure when many substructures (joint association events of target subsets) are repeated to account for the same remaining association possibilities provided associations already made. This can be observed in Figure 3.1 as, departing from the third level in the tree, at which point hypotheses for the targets T1-T3 have already been made, there are 15 identical subtrees that consider either a no-measurement

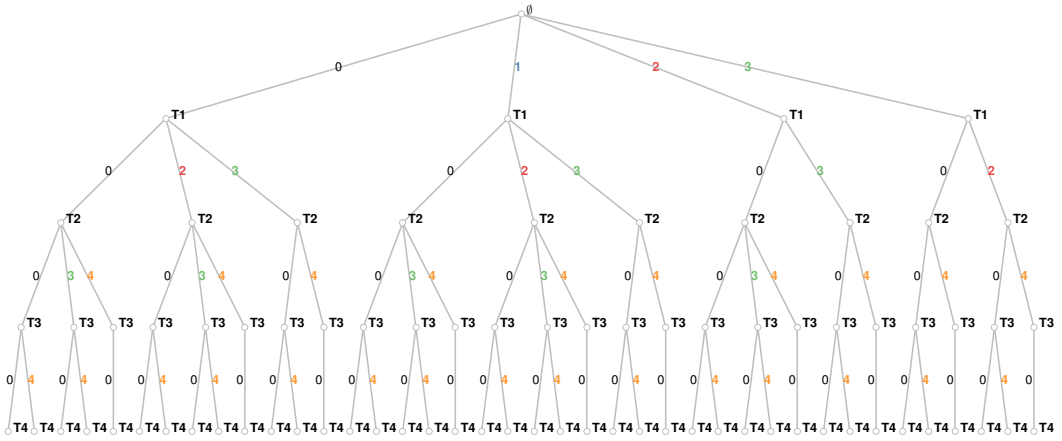


Figure 3.1: Tree of association hypotheses

hypothesis, M0, or measurement M4 to be associated with target T4.

In order to eliminate the association redundancies, the EHM creates a new structure: a probabilistic (acyclic) graph that synthesizes groups of valid associations. In this new structure, given a group of association nodes created for the ith target, the EHM algorithm enumerates valid and unambiguous groups of associations that are possible for the $n - i$ remaining targets. The resulting structure is a network that, during its construction, merges hypotheses that share the same subset of possible association events for targets yet to be analyzed. This constrasts with the standard JPDA approach that enumerates all valid associations of the $(i + 1)$th target for each of the joint hypotheses (association paths) generated for targets up to i. The classical enumeration process (Figure 3.1) renders exponential grow of the number of hypotheses as a function of the number of targets (and measurements), whereas the EHM net captures associations in a much less complex structure.

The computational saving becomes clear when we note that calculating marginal association probabilities is nothing more than a sequence of operations over the association nodes. Ultimately, the complexity for computing the marginal probabilities is dependent on the number of nodes (#P-complete), which in turn depends on the number of targets and measurements. Figure 3.3 illustrates the EHM network constructed based on the possible associations of Table 3.1, considering the target tree of Figure 3.2. The target tree is built such that each node shares at least one (nonnull) measurement assignment with each of its children (e.g., T3 and T2 share M3, and T3 and T4 share M4), and distinct branches do not share any assignments. For the hypothesis tree of Figure 3.1 the total number of nodes is 80, whereas for the net of Figure 3.3 the number of nodes is 11, which illustrates the computational savings obtained by the EHM approach.

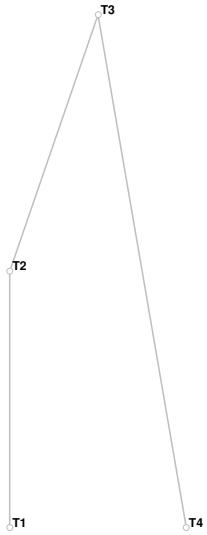


Figure 3.2: Target tree for the association example

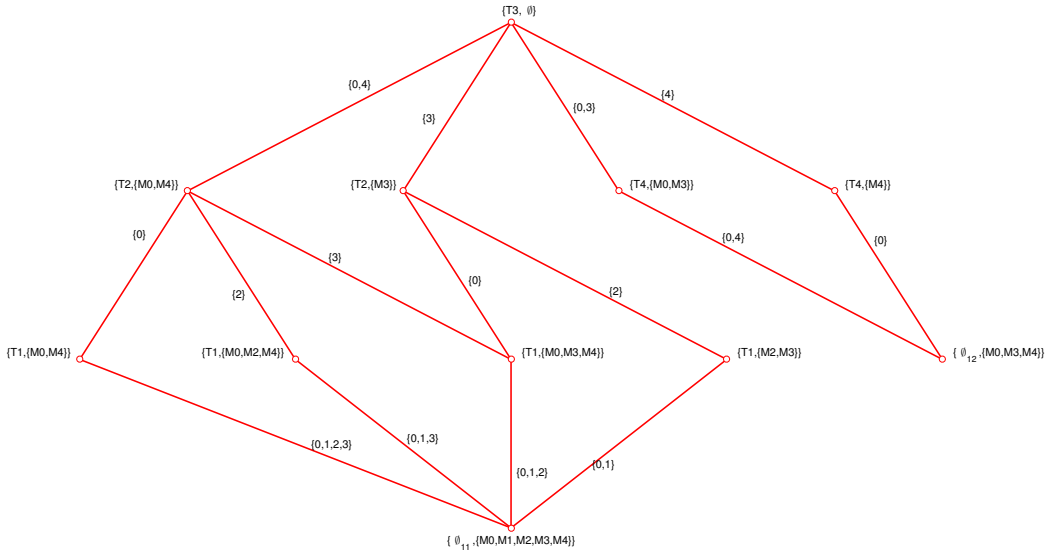


Figure 3.3: EHM net for the association example

Let us describe the construction of the EHM network as follows. As per *Assumption 3.6,* all targets, indexed by $\mathcal{I}_n = \{1, \ldots, n\}$, are arranged in a target tree structure $G_T = (\mathcal{V}_T, \mathcal{E}_T)$ where $\mathcal{V}_T = \mathcal{I}_n$ is the set of vertices and $\mathcal{E}_T = \{(u, v) | u, v \in \mathcal{V}_T, u \neq v\}$ is a set of vertice pairs that describes the edges (e.g., Figure 3.2). All valid measurements are indexed by $\mathcal{I}_{m,0} = \{0\} \cup \{1, 2, \ldots, m\}$, and recall the single association operator $a_k^{(i)} : \mathcal{I}_n \to \mathcal{I}_{m,0}$, which makes the correspondence of each target $i \in \mathcal{I}_n$ to a single measurement $j = a_k^{(i)} \in \mathcal{I}_{m,0}$, where $a_k^{(i)} = 0$ means that the ith target generates no measurement. The EHM network will constitute another graph[2], $G_E = (\mathcal{V}_E, \mathcal{E}_E)$, whose nodes (vertices) in $\mathcal{V}_E$ store information of which measurements have been assigned to the targets up to the node under consideration, and whose edges in $\mathcal{E}_E$

[2]We use the subscripts T and E when denoting the graphs, with T standing for target *tree* and E standing for *EHM*.

store measurements that identify hypotheses made in each parent-child relationship.

We define the set of validated measurements for a given target i as

$$\mathcal{M}_i \triangleq \{\mathrm{y}_k | \mathrm{y}_k \in \mathscr{V}_{k,i}(\tau_{\mathrm{val}})\}, \tag{3.44}$$

where $\mathscr{V}_{k,i}$ is the validation region for the ith target, defined by

$$\mathscr{V}_{k,i}(\tau_{\mathrm{val}}) \triangleq \bigcup_{c_{k-1}^{(i)}} \mathscr{V}_{k,c_{k-1}^{(i)}}(\tau_{\mathrm{val}}), \tag{3.45}$$

$$\mathscr{V}_{k,c_{k-1}^{(i)}}(\tau_{\mathrm{val}}) \triangleq \{\mathrm{y}_k | (\mathrm{y}_k - \hat{\mathrm{y}}_{k|c_{k-1}}^{(i)})^T \mathrm{S}_{k|c_{k-1}}^{(i)-1} (\mathrm{y}_k - \hat{\mathrm{y}}_{k|c_{k-1}}^{(i)}) \leq \tau_{\mathrm{val}}\}, \tag{3.46}$$

where $\mathrm{y}_k - \hat{\mathrm{y}}_{k|c_{k-1}}^{(i)}$ is the innovation vector and $\mathrm{S}_{k|c_{k-1}}^{(i)}$ is the innovation covariance for the target state component $c_{k-1}^{(i)}$, and $\tau_{\mathrm{val}} = \chi_{\mathrm{cdf}}^{-2}(p_g, d_y)$ is the validation threshold, computed based on the chi-square inverse cumulative distribution function for a confidence level (gating probability) p_g and d_y degrees of freedom.

The EHM network is constructed on top of the target tree topology[3]. It starts from the root target in G_T and each target $i \in \mathcal{V}_T$ is processed in turn. The processing sequence follows the target tree structure, exploring each tree level across all descendents of previously processed targets, over all possible branches. For a target i under consideration, the set of all measurements possible to be assigned to targets yet to be evaluated in a branch of G_T, including the current target, is defined as

$$\mathcal{M}_{\{i\}\cup \mathrm{de}(i)} \triangleq \bigcup_{\ell \in \{i\}\cup \mathrm{de}(i)} \mathcal{M}_\ell. \tag{3.47}$$

As it is customary in probabilistic graphical models, conditional independence enables inference in an efficient way. In the EHM context, a target tree is built such that for any two targets $v, w \in G_T$, $v \neq w$, with a common ancestor $u \in \mathrm{an}(v) \cap \mathrm{an}(w)$:

$$\begin{aligned} v \perp\!\!\!\perp w \,|\, u, \\ \therefore \mathcal{M}_{\{v\}\cup \mathrm{de}(v)} \cap \mathcal{M}_{\{w\}\cup \mathrm{de}(w)} = \emptyset, \end{aligned} \tag{3.48}$$

that is, separate branches in the target tree have mutually exclusive possible assignments. For each target $i \in G_T$, a subset of corresponding nodes $\mathcal{N}_i \triangleq \{N_\ell^{(i)} | \ell \in \mathbb{N}\} \in G_E$ will be created. Each node $N^{(i)}$ in the EHM network will pertain to one such node subset corresponding to the ith target. We define the *direct assignment* of a measurement $j \in \mathcal{I}_{m,0}$ to a node $N^{(i)} \in \mathcal{N}_i$, as an attribution $\mathfrak{a} : \mathcal{M}_i \cup \{0\} \to \mathcal{N}_i$ that maps a measurement j to a node $N^{(i)}$ according to $\mathfrak{a}(j) \coloneqq N^{(i)}$, representing the association hypothesis of j being originated from the target i to which $N^{(i)}$ corresponds. The definition of the set of measurements *directly assigned* to a given node $N^{(i)}$ is

$$\mathcal{M}^{\mathfrak{a}}[N^{(i)}] \triangleq \{j \in \mathcal{I}_{m,0} | \mathfrak{a}(j) = N^{(i)}\}. \tag{3.49}$$

In addition, it is convenient to write the set of measurements directly assigned to a node $N^{(i)}$

[3]Note the topological similarity between the target tree of Figure 3.2 and the EHM net of Figure 3.3.

or to any of its ancestors in G_E as

$$\mathcal{M}^{\mathfrak{a}}[\mathrm{an}(N^{(i)}) \cup N^{(i)}] = \left(\bigcup_{N \in \mathrm{an}(N^{(i)})} \mathcal{M}^{\mathfrak{a}}[N] \right) \cup \mathcal{M}^{\mathfrak{a}}[N^{(i)}], \tag{3.50}$$

where $\mathrm{an}\,(N)$ denotes the set of ancestors of a node $N \in \mathcal{V}_E$. Each node $N^{(i)}$ of the EHM network will be uniquely identified by a set composed of its corresponding target i, along with the set of all measurements directly assigned to the node or to any of its ancestors:

$$N^{(i)} \triangleq \{i, \mathcal{M}^{\mathfrak{a}}[\mathrm{an}(N^{(i)}) \cup N^{(i)}]\}. \tag{3.51}$$

The edges of the EHM network represent the parent-child relationship between nodes from different node subsets. Each edge, hereafter denoted as $\mathcal{E}_E[N_r^{(u)}, N_s^{(v)}]$, for any two different targets $u, v \in \mathcal{V}_T$ where $v \in \mathrm{ch}(u)$ (in G_T), is identified by the set of measurements that have been directly assigned to a child node, $N_s^{(v)}$, with index $s \in \{1, \ldots, |\mathcal{N}_v|\}$, under the parenthood of $N_r^{(u)}$, with index $r \in \{1, \ldots, |\mathcal{N}_u|\}$, i.e.,

$$\mathcal{E}_E[N_r^{(u)}, N_s^{(v)}] \triangleq \{\mathcal{M}^{\mathfrak{a}}[N_s^{(v)}] | N_s^{(v)} \in \mathrm{ch}(N_r^{(u)})\}, \tag{3.52}$$

where $\mathrm{ch}\,(N)$ denotes the set of children of node N. The children of a given node $N^{(u)}$ will be a subset of nodes corresponding to the next target to be processed, $v \in \mathrm{ch}(u)$ (in G_T), that is, $\mathrm{ch}(N^{(u)}) \subseteq \mathcal{N}_v$ (in G_E), where $\mathcal{N}_v = \{N_\ell^{(v)} | \ell \in \mathbb{N}\}$. This is because once the previous node subset $\mathcal{N}_u$ is complete, the current node subset, $\mathcal{N}_v$, shall cover hypotheses for the target v given the associations already made for all targets down to target u in a branch of G_T. Given a node $N^{(v)}$, the available measurements possible to be assigned to its descendents, corresponding to the remaining targets in the branch, can be described by the set

$$\mathcal{M}^{\mathrm{avail}}_{\mathrm{de}(v)}[N^{(v)}] = \mathcal{M}_{\mathrm{de}(v)} \backslash \left(\mathcal{M}^{\mathfrak{a}}[\mathrm{an}(N^{(v)}) \cup N^{(v)}] \backslash \{0\} \right). \tag{3.53}$$

In order to construct the node subset $\mathcal{N}_v$ under parenthood of nodes previously constructed, each node in $\mathcal{N}_u$ is considered in turn as a potential parent node. For each node $N^{(u)} \in \mathcal{N}_u$, every possible candidate measurement $j_c \in \mathcal{M}_v$ that can be assigned to the target being processed, v, is considered. For each tentative pair $(N^{(u)}, j_c) | j_c \in \mathcal{M}_v$, the available measurements possible to be assigned to the remaining targets in the branch, excluding the measurements already assigned to the node $N^{(u)}$ or to its ancestors, is described by the set

$$\mathcal{M}^{\mathrm{avail}}_{\mathrm{de}(u)}[N^{(u)}, j_c] = \mathcal{M}_{\mathrm{de}(u)} \backslash \left(\mathcal{M}^{\mathfrak{a}}[\mathrm{an}(N^{(u)}) \cup N^{(u)}] \cup \{j_c\} \backslash \{0\} \right). \tag{3.54}$$

Notice that the dummy measurement, $j_c = 0$, always remains as a possibility for any of the upcoming associations: this is why it is never excluded from the set of available measurements in (3.53) or (3.54). The candidate measurements to establish associations are taken from the set of validated measurements for the target under analysis, i.e., $\mathcal{M}_v$. Each candidate measurement that is associated to the target v is then removed from the current set of candidates, and is directly assigned to a certain node in $\mathcal{N}_v$ (under the parenthood of a node in $\mathcal{N}_u$).

The set $\mathcal{N}_u$ is complete when all candidate measurements for target u have been assigned to nodes $N^{(u)} \in \mathcal{N}_u$. Provided a target $v \in \mathrm{ch}(u)$ (in G_T), in order to start a new node subset,

assuming $\mathcal{N}_v = \emptyset$ initially, a new node is created as $N_1^{(v)} \coloneqq \{v, \mathcal{M}^{\mathfrak{a}}[\text{an}(N_1^{(u)}) \cup N_1^{(u)}] \cup \{j_{c,1}\}\}$, including the first candidate target, $j_{c,1} \in \mathcal{M}_v$, and identified as a child of the first node $N_1^{(u)} \in \mathcal{N}_u$. As a consequence, the parent-child relationship between $N_1^{(u)}$ and $N_1^{(v)}$ is identified by the measurement $j_{c,1}$, which is directly assigned to $N_1^{(v)}$ under parenthood of $N_1^{(u)}$, and so $\mathcal{E}_E[N_1^{(u)}, N_1^{(v)}] \coloneqq \{j_{c,1}\}$. For creating new nodes in $\mathcal{N}_v$, each potential parent node $N_r^{(u)} \in \mathcal{N}_u$, where $r \in \{1, \ldots, |\mathcal{N}_u|\}$, is verified in turn, based on which all nodes already included in $\mathcal{N}_v$ are analyzed. If any existing node, $N_s^{(v)} \in \mathcal{N}_v$, where $s \in \{1, \ldots, |\mathcal{N}_v|\}$, has the same set of remaining available measurements as that of the candidate measurement j_c under parenthood of $N_r^{(u)}$, i.e. $\mathcal{M}^{\text{avail}}_{\text{de}(v)}[N_s^{(v)}] = \mathcal{M}^{\text{avail}}_{\text{de}(u)}[N_r^{(u)}, j_c]$, then the parent-child relationship between $N_r^{(u)}$ and $N_s^{(v)}$ is acknowledged and its identification incorporates the candidate measurement. This is done by updating the sets that identify $N_s^{(v)}$ and $\mathcal{E}_E[N_r^{(u)}, N_s^{(v)}]$ to include $\{j_c\}$, and no new node is added. Otherwise, if none of the existing nodes in $\mathcal{N}_v$ has the same set of remaining available targets as that of the candidate measurement j_c under parenthood of $N_r^{(u)}$, then a new child node is created as $N_s^{(v)} \coloneqq \{v, \mathcal{M}^{\mathfrak{a}}[\text{an}(N_r^{(u)}) \cup N_r^{(u)}] \cup \{j_c\}\}$, with $s = |\mathcal{N}_v| + 1$, such that the parent-child relationship between $N_r^{(u)}$ and $N_s^{(v)}$ is identified as the direct assignment $\mathfrak{a}(j_c) \coloneqq N_s^{(v)}$ is made and $\mathcal{E}_E[N_r^{(u)}, N_s^{(v)}] \coloneqq \{j_c\}$. The network is complete when all targets in all levels of the target tree have been processed and no available measurements for association remain.

3.5.2 Inference via EHM

For simplicity of exposition, we first describe a case where each target state has a single component, and then we extend it to the case where each target state is described by multiple components. EHM eases computation of marginal association probabilities, $p(a_k^{(i)}|\text{Y}_{1:k})$, over the set of valid joint association events, $a_k^{(1:n)} \in \mathcal{A}_{\mathcal{I}_n,k}$, encoded in the EHM network. The computation of marginal association probabilities is formulated as a recursive sequence of operations over the EHM net nodes. The formulation follows the exposition in [95]. Denoting the event $A^{\mathcal{S}} \coloneqq \{a_k^{\mathcal{S}} \in \mathcal{A}_{\mathcal{S},k} | \mathcal{S} \subseteq \mathcal{I}_n\}$, the intent is to calculate a marginalization of the form

$$p(a_k^{(i)} = j|\text{Y}_{1:k}) = \sum_{a_k^{(1:n)} \setminus a_k^{(i)} | a_k^{(i)} = j} p(a_k^{(1:n)}|A^{(1:n)}, \text{Y}_{1:k}), \tag{3.55}$$

$$p(a_k^{(i)} = j|\text{Y}_{1:k}) \propto \sum_{a_k^{(i)} = j} \mathbb{1}_{\mathcal{A}_{\mathcal{I}_n,k}}(a_k^{(1:n)}) \prod_{\ell=1}^{n} \psi_\ell(a_k^{(\ell)}), \tag{3.56}$$

where

$$\psi_\ell(j) = \begin{cases} \lambda_{V,\text{fa}}(1 - p_d), & j = 0, \\ p_d \int_{\mathcal{X}} \ell_j(\text{x}_k^{(\ell)}) p(\text{x}_k^{(\ell)}|\text{Y}_{1:k-1}) d\text{x}_k^{(\ell)}, & j \neq 0, \end{cases} \tag{3.57}$$

and where we assumed $p_d(\text{x}_k) = p_d$. Because the EHM net encodes $A^{(1:n)} = \{a_k^{(1:n)} \in \mathcal{A}_{\mathcal{I}_n,k}\}$, (3.56) can be computed based on factors over the net G_E. Note that $a_k^{(i),\text{an}(i),\text{de}(i),\text{cs}(i)} \equiv a_k^{(1:n)}$

and (3.55) can be factored as

$$
\begin{aligned}
p(a_k^{(i)}|\mathrm{Y}_{1:k}) &\propto \sum_{a_k^{(1:n)}\setminus a_k^{(i)}} p(a_k^{(i),\mathrm{an}(i),\mathrm{de}(i),\mathrm{cs}(i)}|A^{(1:n)},\mathrm{Y}_{1:k}) \\
= \sum_{a_k^{\mathrm{an}(i)}} \sum_{a_k^{(1:n)}\setminus a_k^{(i),\mathrm{an}(i)}} & p(a_k^{\mathrm{de}(i)}|a_k^{(i),\mathrm{an}(i)},A^{(1:n)},\mathrm{Y}_{1:k}) p(a_k^{(i)}|a_k^{\mathrm{an}(i)},A^{(1:n)},\mathrm{Y}_{1:k}) p(a_k^{\mathrm{an}(i),\mathrm{cs}(i)}|A^{(1:n)},\mathrm{Y}_{1:k}) \\
&= \sum_{a_k^{\mathrm{an}(i)}} \left(\sum_{a_k^{\mathrm{de}(i)}} p(a_k^{\mathrm{de}(i)}|a_k^{(i),\mathrm{an}(i)},A^{(1:n)},\mathrm{Y}_{1:k}) \right) \\
&\quad \times p(a_k^{(i)}|a_k^{\mathrm{an}(i)},A^{(1:n)},\mathrm{Y}_{1:k}) \left(\sum_{a_k^{\mathrm{cs}(i)}} p(a_k^{\mathrm{an}(i),\mathrm{cs}(i)}|A^{(1:n)},\mathrm{Y}_{1:k}) \right). \quad (3.58)
\end{aligned}
$$

For $v, w \in \mathrm{ch}(u)$, $v \perp\!\!\!\perp w \,|\, u$, and so

$$
\begin{aligned}
\sum_{a_k^{\mathrm{de}(i)}} p(a_k^{\mathrm{de}(i)}|a_k^{(i),\mathrm{an}(i)},A^{(1:n)},\mathrm{Y}_{1:k}) &= \sum_{a_k^{\mathrm{de}(i)}} p(a_k^{\mathrm{ch}(i),\mathrm{de}(\mathrm{ch}(i))}|a_k^{(i),\mathrm{an}(i)},A^{(1:n)},\mathrm{Y}_{1:k}) \\
&= \sum_{a_k^{\mathrm{de}(i)}} \prod_{\ell\in\mathrm{ch}(i)} p(a_k^{(\ell),\mathrm{de}(\ell)}|a_k^{\mathrm{an}(\ell)},A^{(1:n)},\mathrm{Y}_{1:k}) \\
&= \prod_{\ell\in\mathrm{ch}(i)} \sum_{a_k^{(\ell),\mathrm{de}(\ell)}} p(a_k^{(\ell),\mathrm{de}(\ell)}|a_k^{\mathrm{an}(\ell)},A^{(1:n)},\mathrm{Y}_{1:k}),
\end{aligned}
$$

since the marginal on a subtree over descendents of i is the product of marginals on its mutually exclusive branches (one branch per child of i). By defining

$$
\begin{aligned}
\breve{w}_B(i,a_k^{\mathrm{an}(i)}) &\triangleq \sum_{a_k^{(i),\mathrm{de}(i)}} p(a_k^{(i),\mathrm{de}(i)}|a_k^{\mathrm{an}(i)},A^{(1:n)},\mathrm{Y}_{1:k}) \\
&= \sum_{a_k^{(i),\mathrm{de}(i)}} \prod_{\ell\in\{i\}\cup\mathrm{de}(i)} \mathbb{1}_{\breve{\mathcal{A}}_{\ell,k}}(a_k^{(\ell),\mathrm{an}(\ell)})\psi_\ell(a_k^{(\ell)}), \quad (3.59) \\
\breve{w}_F(i,a_k^{\mathrm{an}(i)}) &\triangleq \sum_{a_k^{\mathrm{cs}(i)}} p(a_k^{\mathrm{an}(i),\mathrm{cs}(i)}|A^{(1:n)},\mathrm{Y}_{1:k}) \\
&= \sum_{a_k^{\mathrm{cs}(i)}} \prod_{\ell\in\mathrm{an}(i)\cup\mathrm{cs}(i)} \mathbb{1}_{\breve{\mathcal{A}}_{\ell,k}}(a_k^{(\ell),\mathrm{an}(\ell)})\psi_\ell(a_k^{(\ell)}), \quad (3.60)
\end{aligned}
$$

then

$$
p(a_k^{(i)}|\mathrm{Y}_{1:k}) = \sum_{a_k^{\mathrm{an}(i)}} \left(\prod_{\ell\in\mathrm{ch}(i)} \breve{w}_B(\ell,a_k^{\mathrm{an}(\ell)}) \right) \cdot \mathbb{1}_{\breve{\mathcal{A}}_{i,k}}(a_k^{(i),\mathrm{an}(i)})\psi_i(a_k^{(i)}) \cdot \breve{w}_F(i,a_k^{\mathrm{an}(i)}). \quad (3.61)
$$

Rewritting (3.61) in terms of factors distributed over the EHM network is possible because each node $N^{(i)} \in G_E$ possesses information on associations made for the corresponding target $i \in G_T$ and its ancestors. This is clear from the event equivalence: $\{j \in N^{(i)}\} \equiv \{j \in a_k^{(i),\mathrm{an}(i)}\}$.

Therefore, the factorization in (3.61) can be written as [95]

$$p(a_k^{(i)} = j|\mathrm{Y}_{1:k}) \propto \sum_{N^{(i)} \in \mathcal{N}_i} p(a_k^{(i)} = j, N^{(i)}|\mathrm{Y}_{1:k}) \triangleq \sum_{N^{(i)} \in \mathcal{N}_i} \pi(j, N^{(i)}), \tag{3.62}$$

where $\pi(j, N)$ is the joint likelihood of all paths over the net passing through node N and satisfying the assignment $\mathfrak{a}(j) = N$, i.e. $j \in N$. For each association $j \coloneqq a_k^{(i)}$, the likelihood $\pi(j, N^{(i)})$ is a quantity calculated over nodes $N^{(i)} \in \mathcal{N}_i$ such that $N^{(i)} \ni j$. In view of (3.61), $\pi(j, N^{(i)})$ can be written as

$$\pi(j, N^{(i)}) = \psi_{\mathrm{de}(i)}(j, N^{(i)}) \cdot \psi_i(j) \cdot \psi_{\overline{\mathrm{de}}(i)}(j, N^{(i)}) \tag{3.63}$$

$$= \prod_{N^{(c)} \in \mathcal{N}_{\mathrm{ch}(i)}} w_B(N^{(c)})^{\mathbb{1}_{\mathcal{E}_E[N^{(i)}, N^{(c)}]}(j)} \cdot \psi_i(j) \cdot \mathbb{1}_{\mathcal{E}_E[N^{(i)}, N^{\mathrm{ch}(i)}]}(j)\, w_F(N^{(i)}), \tag{3.64}$$

where we use the notation $\mathcal{N}_{\mathcal{S}} = \bigcup_{\ell \in \mathcal{S}} \mathcal{N}_\ell$ for any subset $\mathcal{S} \subseteq \mathcal{I}_n$. In (3.64), observe the indicator function $\mathbb{1}_{\mathcal{E}_E[N^{(i)}, N^{(c)}]}(j)$ as an exponent of the product terms so that only backward factors under assignment of j take effect, and we use the indicator function $\mathbb{1}_{\mathcal{E}_E[N^{(i)}, N^{\mathrm{ch}(i)}]}(j)$, where $\mathcal{E}_E[N^{(i)}, N^{\mathrm{ch}(i)}] \coloneqq \bigcup_{N^{(c)} \in \mathcal{N}_{\mathrm{ch}(i)}} \mathcal{E}_E[N^{(i)}, N^{(c)}]$, to account for a forward factor only if $N^{(i)}$ has parenthood under the assignment of j to one of its children. The following factor definitions were made:

$$\psi_{\mathrm{de}(i)}(j, N^{(i)}) \coloneqq \prod_{N^{(c)} \in \mathcal{N}_{\mathrm{ch}(i)}} w_B(N^{(c)})^{\mathbb{1}_{\mathcal{E}_E[N^{(i)}, N^{(c)}]}(j)}, \tag{3.65}$$

$$\psi_{\overline{\mathrm{de}}(i)}(j, N^{(i)}) \coloneqq \mathbb{1}_{\mathcal{E}_E[N^{(i)}, N^{\mathrm{ch}(i)}]}(j)\, w_F(N^{(i)}), \tag{3.66}$$

where $w_B(\cdot)$ are backward weights accounting for the probabilistic effect of all descendents of a node (containing measurement j), $\psi_i(\cdot)$ is the single association likelihood for target i, and $w_F(\cdot)$ are forward weights accounting for the effect from non-descendants of a node. The backward and forward weights accumulate probabilistic effect on a node $N^{(i)}$ from other nodes in G_E, and can be calculated recursively by the forward-backward algorithm as

$$w_B(N^{(i)}) = \sum_{j \in \mathcal{M}_{\mathrm{ch}(i)}} \psi_i(j) \cdot \prod_{N^{(c)} \in \mathcal{N}_{\mathrm{ch}(i)}} w_B(N^{(c)})^{\mathbb{1}_{\mathcal{E}_E[N^{(i)}, N^{(c)}]}(j)}, \tag{3.67}$$

$$w_F(N^{(i)}) = \textstyle\sum_{N^{(p)} \in \mathcal{N}_{\mathrm{pa}(i)}} \sum_{j \in \mathcal{M}_{\mathrm{ch}(p) \setminus \{i\}}} \psi_p(j) \left(\prod_{N^{(s)} \in \mathcal{N}_{\mathrm{ch}(p) \setminus \{i\}}} w_B(N^{(s)})^{\mathbb{1}_{\mathcal{E}_E[N^{(p)}, N^{(s)}]}(j)} \right) w_F(N^{(p)}), \tag{3.68}$$

where $\mathcal{M}_{\mathcal{S}} = \bigcup_{\ell \in \mathcal{S}} \mathcal{M}_\ell$ for any subset $\mathcal{S} \subseteq \mathcal{I}_n$, and it is clear the correspondence of (3.67) to (3.59), and (3.68) to (3.60). The forward weight of the net root and the backward weight of the deepest leaves are initialized as $w_F(\{\cdot, \emptyset\}) = w_B(\{\emptyset, \cdot\}) \coloneqq 1$. This scheme provides exact solution for inference problems in trees and other acyclic graphs (e.g., polytrees) such as the EHM network.

The EHM algorithm was originally proposed in a simpler form [135] where a simple list of targets is considered, but featuring heuristic target ordering with the intent to minimize the number of nodes. A second version of the algorithm (EHM 2) was proposed in [95], incorporating the target tree structure and clustering of independent groups of targets, each giving rise to a different EHM net. The EHM 2 is akin to the algorithm presented in this section, where each group of targets does not share measurements with other groups and its target tree is

based on the rule that each parent target shares at least one validated measurement with each of its children. The third variation, EHM 3, was proposed in [94] to incorporate inference of marginal associations for multiple components in target state mixtures, where each component can possibly depend on components of other targets.

3.5.3 Marginal association probabilities with EHM 3

Generalizing the procedure to derive the EHM 3 used in this chapter requires a factorization very similar to (3.61) applied to $p(c_{k-1}^{(1:n)}, a_k^{(1:n)}, E_k^{\mathcal{I}_{d,k}}|, \mathrm{Y}_{1:k})$, including marginalizations over mixture components. The intermediate expressions are tedious and can be found in [95, 94], however we believe the argument provided in the previous section should be enough to induce the reader to derive the result. For simplifying the notation we will omit conditioning on the joint existence event, $E_k^{\mathcal{I}_{d,k}}$, by recalling the event equivalence $(a_k^{(1:n)}, E_k^{\mathcal{I}_{d,k}}) \equiv a_k^{(1:n)}$, and we denote $\psi(a_k^{(i)}|c_{k-1}^{(i)}) \coloneqq \psi(a_k^{(i)}, E_k^{(i)}|c_{k-1}^{(i)})$ according to (3.32). Generalizing the result of (3.62), (3.67), and (3.68) for the EHM 3, conditioned on target existence, we obtain

$$
\begin{aligned}
&p(c_{k-1}^{(i)}, a_k^{(i)}|\mathrm{Y}_{1:k}) \\
&\quad = \sum_{a_k^{\mathrm{an}(i)}} \sum_{a_k^{(1:n)} \setminus a_k^{(i),\mathrm{an}(i)}} \sum_{c_{k-1}^{(1:n)} \setminus c_{k-1}^{(i)}} p(c_{k-1}^{\mathrm{de}(i)}, a_k^{\mathrm{de}(i)}|c_{k-1}^{(i)}, a_k^{(i),\mathrm{an}(i)}, A^{(1:n)}, \mathrm{Y}_{1:k}) \\
&\qquad\qquad \times p(c_{k-1}^{(i)}, a_k^{(i)}|a_k^{\mathrm{an}(i)}, A^{(1:n)}, \mathrm{Y}_{1:k}) p(c_{k-1}^{\mathrm{an}(i),\mathrm{cs}(i)}, a_k^{\mathrm{an}(i),\mathrm{cs}(i)}|A^{(1:n)}, \mathrm{Y}_{1:k}),
\end{aligned}
$$

$$
p(c_{k-1}^{(i)}, a_k^{(i)}|\mathrm{Y}_{1:k}) \propto \sum_{N^{(i)} \in \mathcal{N}_i} \pi(c_{k-1}^{(i)}, a_k^{(i)}, N^{(i)}), \tag{3.69}
$$

where

$$
\begin{aligned}
\pi(c_{k-1}^{(i)}, a_k^{(i)}, N^{(i)}) = &\left(\prod_{N^{(c)} \in \mathcal{N}_{\mathrm{ch}(i)}} w_B(N^{(c)}, c_{k-1}^{(i)})^{\mathbb{1}_{\mathcal{E}_E[N^{(i)}, N^{(c)}]}(a_k^{(i)})} \right) \cdot \psi(a_k^{(i)}|c_{k-1}^{(i)}) \\
&\times \sum_{c_{k-1}^{\mathrm{pa}(i)} \in \Gamma_{k-1}^{\mathrm{pa}(i)}} \mathbb{1}_{\mathcal{E}_E[N^{(i)}, N^{\mathrm{ch}(i)}]}(a_k^{(i)})\, p(c_{k-1}^{(i)}|c_{k-1}^{\mathrm{pa}(i)}, \mathrm{Y}_{1:k-1}) w_F(N^{(i)}, c_{k-1}^{\mathrm{pa}(i)}).
\end{aligned} \tag{3.70}
$$

The backward and forward weights are computed as

$$w_B(N^{(i)}, c_{k-1}^{\mathrm{pa}(i)}) = \sum_{j \in \mathcal{M}_{\mathrm{ch}(i)}} \sum_{c_{k-1}^{(i)} \in \Gamma_{k-1}^{(i)}} \psi(j|c_{k-1}^{(i)}) p(c_{k-1}^{(i)}|c_{k-1}^{\mathrm{pa}(i)}, \mathrm{Y}_{1:k-1}) \times \prod_{N^{(c)} \in \mathcal{N}_{\mathrm{ch}(i)}} w_B(N^{(c)}, c_{k-1}^{(i)})^{\mathbb{1}_{\mathcal{E}_{E[N^{(i)}, N^{(c)}]}}(j)}, \tag{3.71}$$

$$\begin{aligned} w_F(N^{(i)}, c_{k-1}^{\mathrm{pa}(i)}) = & \sum_{N^{(p)} \in \mathcal{N}_{\mathrm{pa}(i)}} \sum_{j \in \mathcal{M}_{\mathrm{ch}(p)\setminus\{i\}}} \psi(j|c_{k-1}^{(p)}) \\ & \times \left(\prod_{N^{(s)} \in \mathcal{N}_{\mathrm{ch}(p)\setminus\{i\}}} w_B(N^{(s)}, c_{k-1}^{(p)})^{\mathbb{1}_{\mathcal{E}_{E[N^{(p)}, N^{(s)}]}}(j)} \right) \\ & \times \sum_{c_{k-1}^{\mathrm{pa}(p)} \in \Gamma_{k-1}^{\mathrm{pa}(p)}} p(c_{k-1}^{(p)}|c_{k-1}^{\mathrm{pa}(p)}, \mathrm{Y}_{1:k-1}) w_F(N^{(p)}, c_{k-1}^{\mathrm{pa}(p)}). \end{aligned} \tag{3.72}$$

Finally, the pairwise marginal association probabilities can be derived as

$$\begin{aligned} & p(c_{k-1}^{(i),\mathrm{pa}(i)}, a_k^{(i),\mathrm{pa}(i)}|\mathrm{Y}_{1:k}) \\ = & \sum_{a_k^{\mathrm{an}(\mathrm{pa}(i))}} \sum_{a_k^{(1:n)} \setminus a_k^{(i),\mathrm{pa}(i),\mathrm{an}(\mathrm{pa}(i))}} \sum_{c_{k-1}^{(1:n)} \setminus c_{k-1}^{(i),\mathrm{pa}(i)}} p(c_{k-1}^{\mathrm{de}(i)}, a_k^{\mathrm{de}(i)}|c_{k-1}^{(i),\mathrm{pa}(i)}, a_k^{(i),\mathrm{pa}(i),\mathrm{an}(\mathrm{pa}(i))}, A^{(1:n)}, \mathrm{Y}_{1:k}) \\ & \times p(c_{k-1}^{(i),\mathrm{pa}(i)}, a_k^{(i),\mathrm{pa}(i)}|a_k^{\mathrm{an}(\mathrm{pa}(i))}, A^{(1:n)}, \mathrm{Y}_{1:k}) \\ & \times p(c_{k-1}^{\mathrm{an}(\mathrm{pa}(i)),\mathrm{cs}(i),\mathrm{cs}(\mathrm{pa}(i))}, a_k^{\mathrm{an}(\mathrm{pa}(i)),\mathrm{cs}(i),\mathrm{cs}(\mathrm{pa}(i))}|A^{(1:n)}, \mathrm{Y}_{1:k}), \end{aligned}$$

$$\therefore p(c_{k-1}^{(i),\mathrm{pa}(i)}, a_k^{(i),\mathrm{pa}(i)}|\mathrm{Y}_{1:k}) \propto \sum_{N^{\mathrm{pa}(i)} \in \mathcal{N}_{\mathrm{pa}(i)}} \pi(c_{k-1}^{(i),\mathrm{pa}(i)}, a_k^{(i),\mathrm{pa}(i)}, N^{\mathrm{pa}(i)}), \tag{3.73}$$

where

$$\begin{aligned} & \pi(c_{k-1}^{(i),\mathrm{pa}(i)}, a_k^{(i),\mathrm{pa}(i)}, N^{\mathrm{pa}(i)}) = \\ & \left(\prod_{N^{(c)} \in \mathcal{N}_{\mathrm{ch}(i)}} w_B(N^{(c)}, c_{k-1}^{(i)})^{\mathbb{1}_{\mathcal{E}_{E[N^{(i)}, N^{(c)}]}}(a_k^{(i)})} \right) \psi(a_k^{(i)}|c_{k-1}^{(i)}) p(c_{k-1}^{(i)}|c_{k-1}^{\mathrm{pa}(i)}, \mathrm{Y}_{1:k-1}) \psi(a_k^{\mathrm{pa}(i)}|c_{k-1}^{\mathrm{pa}(i)}) \\ & \times \sum_{c_{k-1}^{\mathrm{pa}(\mathrm{pa}(i))} \in \Gamma_{k-1}^{\mathrm{pa}(\mathrm{pa}(i))}} p(c_{k-1}^{\mathrm{pa}(i)}|c_{k-1}^{\mathrm{pa}(\mathrm{pa}(i))}, \mathrm{Y}_{1:k-1}) w_F(N^{\mathrm{pa}(i)}, c_{k-1}^{\mathrm{pa}(\mathrm{pa}(i))}) \\ & \times \left(\prod_{N^{(s)} \in \mathcal{N}_{\mathrm{sb}(i)}} w_B(N^{(s)}, c_{k-1}^{\mathrm{pa}(i)})^{\mathbb{1}_{\mathcal{E}_{E[N^{\mathrm{pa}(i)}, N^{(s)}]}}(a_k^{\mathrm{pa}(i)})} \right), \end{aligned} \tag{3.74}$$

where we omitted conditioning on target existence for simplifying the exposition. Finally, following (3.35):

$$p(c_{k-1}^{(i)}, a_k^{(i)}|E_k^{(i)}, \mathrm{Y}_{1:k}) = p(E_k^{(i)}|a_k^{(i)}) \frac{p(c_{k-1}^{(i)}, a_k^{(i)}|\mathrm{Y}_{1:k})}{p(E_k^{(i)}|\mathrm{Y}_{1:k})}, \tag{3.75}$$

$$p(c_{k-1}^{(i),\mathrm{pa}(i)}, a_k^{(i),\mathrm{pa}(i)}|E_k^{(i),\mathrm{pa}(i)}, \mathrm{Y}_{1:k}) = p(E_k^{(i),\mathrm{pa}(i)}|a_k^{(i),\mathrm{pa}(i)}) \frac{p(c_{k-1}^{(i),\mathrm{pa}(i)}, a_k^{(i),\mathrm{pa}(i)}|\mathrm{Y}_{1:k})}{p(E_k^{(i),\mathrm{pa}(i)}|\mathrm{Y}_{1:k})}, \tag{3.76}$$

where $p(E_k^{(i)}|a_k^{(i)}) = \alpha^{\mathcal{I}_{\bar{a},k} \cap \{i\}}$ and $p(E_k^{(i),\mathrm{pa}(i)}|a_k^{(i),\mathrm{pa}(i)}) = \alpha^{\mathcal{I}_{\bar{a},k} \cap \{i,\mathrm{pa}(i)\}}$.

3.6 Construction and update of target trees

As per *Assumption 3.6,* targets take part in one or more independent target trees, which are built based on independent groups of targets. In this section we explain the construction and prediction of each target tree. If no target tree exists at a previous time step, new target trees are built using the EHM 2 subroutine to construct trees, initializing a single-valued state component $c_k^{(i)}$ per node. Otherwise, provided a set of previous target trees $G_{T,k-1}^{\mathcal{I}_{g,k-1}} = \{(\mathcal{V}_{T,k-1}^{(g)}, \mathcal{E}_{T,k-1}^{(g)}) | g \in \mathcal{I}_{g,k-1}\}$, where $\mathcal{I}_{g,k-1}$ indexes all independent target clusters at time step $k-1$, and a set of predicted targets indexed by $\mathcal{I}_{n,k|k-1} = \mathcal{I}_{c,k|k-1} \cup \mathcal{I}_{u,k|k-1}$, before applying the EHM 3, a set of predicted target trees $G_{T,k|k-1}^{\mathcal{I}_{g,k}} = \{(\mathcal{V}_{T,k|k-1}^{(g)}, \mathcal{E}_{T,k|k-1}^{(g)}) | g \in \mathcal{I}_{g,k}\}$ is constructed, identified by the index set $\mathcal{I}_{g,k}$ of target clusters at time step k. Each predicted tree incorporates all targets that may share possible associations, i.e., there must be one predicted tree per target cluster. Target trees are updated after the marginal association probabilities have been computed and inference on new target components have been accomplished.

3.6.1 Initial target trees

During initial filtering steps or periods when no targets remain in the scene, no target tree exists. In those cases, as soon as new measurements are validated for any of the unconfirmed targets, one or more target trees must be built. The procedure is reminiscent of the EHM 2 method for building trees [95]. Targets that share a subset of measurements as possible associations are grouped together in clusters as

$$\mathcal{C}_{k|k-1}^{(g)} \triangleq \{i \in \mathcal{I}_{n,k|k-1}^{(g)} \subseteq \mathcal{I}_{n,k|k-1} | \mathcal{M}_i \cap \mathcal{M}_j \neq \emptyset, \forall i,j \in \mathcal{I}_{n,k|k-1}^{(g)}, i \neq j\},\ g \in \mathbb{N}. \tag{3.77}$$

Each initial target cluster, $\mathcal{C}_{k|k-1}^{(g)} \subseteq \mathcal{I}_{u,k|k-1}$, for some $g \in \mathbb{N}$, will give origin to a target tree $G_{T,k|k-1}^{(g)} = (\mathcal{V}_{T,k|k-1}^{(g)}, \mathcal{E}_{T,k|k-1}^{(g)})$, being processed in turn. Firstly, a target arbitrarily chosen, $i_1 \in \mathcal{C}_{k|k-1}^{(g)}$, is added with no parents and no children ($\mathrm{pa}(i_1) = \mathrm{ch}(i_1) \coloneqq \emptyset$) in a newly initialized tree, $G_{T,k|k-1}^{(g)}$, and its set of accumulated assignments is defined as $\mathcal{M}_{i_1,\mathrm{ch}(i_1)} \triangleq \mathcal{M}_{i_1}$. Then, another target $i_2 \in \mathcal{C}_{k|k-1}^{(g)} \setminus \{i_1\}$ is verified, and its set of possible assignments, $\mathcal{M}_{i_2}$, is compared to accumulated assignments of the node already added, $\mathcal{M}_{i_1,\mathrm{ch}(i_1)}$. If $\mathcal{M}_{i_2} \cap \mathcal{M}_{i_1,\mathrm{ch}(i_1)} \neq \emptyset$, then i_2 is added to the tree as the parent of i_1, by setting $\mathrm{pa}(i_2) \coloneqq \emptyset$, $\mathrm{ch}(i_2) \coloneqq \{i_1\}$, $\mathcal{M}_{i_2,\mathrm{ch}(i_2)} \coloneqq \mathcal{M}_{i_2} \cup \mathcal{M}_{i_1,\mathrm{ch}(i_1)}$, and updating the parent of i_1 as $\mathrm{pa}(i_1) \coloneqq \{i_2\}$. Else, if $\mathcal{M}_{i_2} \cap \mathcal{M}_{i_1,\mathrm{ch}(i_1)} = \emptyset$, then i_2 is simply added with $\mathrm{pa}(i_2) \coloneqq \emptyset$ and no relation with i_1 is established. Next, other targets $i \in \mathcal{C}_{k|k-1}^{(g)} \setminus \{i_1, i_2\}$ are considered in turn, where the set of

possible assignments, $\mathcal{M}_i$, is compared to accumulated assignments of all other nodes that have no parent. Before starting the comparisons, node i is added by setting $\text{pa}(i) \coloneqq \emptyset$, $\text{ch}(i) \coloneqq \emptyset$ and $\mathcal{M}_{i,\text{ch}(i)} \coloneqq \mathcal{M}_i$. Hence, verifying each $\ell \in \mathcal{V}^{(g)}_{T,k|k-1} \setminus \{i\}$ such that $\text{pa}(\ell) = \emptyset$ in turn, if $\mathcal{M}_i \cap \mathcal{M}_{\ell,\text{ch}(\ell)} \neq \emptyset$, then i is updated as the parent of ℓ by making $\text{ch}(i) \leftarrow \text{ch}(i) \cup \{\ell\}$, $\mathcal{M}_{i,\text{ch}(i)} \leftarrow \mathcal{M}_{i,\text{ch}(i)} \cup \mathcal{M}_{\ell,\text{ch}(\ell)}$, $\text{pa}(\ell) \coloneqq \{i\}$. If $\mathcal{M}_i \cap \mathcal{M}_{\ell,\text{ch}(\ell)} = \emptyset$ for all $\ell \in \mathcal{V}^{(g)}_{T,k|k-1} \setminus \{i\}$ with $\text{pa}(\ell) = \emptyset$, then no relation is made. In this procedure the tree is constructed bottom-up, and different branches are guaranteed to have mutually exclusive assignments.

For convenience of calculations, we represent the target tree such that a conditional (discrete) distribution $p(c^{(i)}_{k-1}|c^{\text{pa}(i)}_{k-1}, \text{Y}_{1:k-1})$ (*Assumption 3.6*) is maintained for each edge and a marginal (discrete) distribution $p(c^{(i)}_{k-1}|\text{Y}_{1:k-1})$ is maintained for each node. Each node in the initial target tree is represented by a discrete random variable $c^{(i)}_{k-1}$ that has only one possible value because unconfirmed targets keep only one state component (*Assumption 3.8*). Therefore, an initial target tree is set with $p(c^{(i)}_{k-1}) = 1$ and $p(c^{(i)}_{k-1}|c^{\text{pa}(i)}_{k-1}) = 1$ for all $i \in \mathcal{I}_{u,k|k-1}$, meaning that $p(c^{(i)}_{k-1}|c^{\text{pa}(i)}_{k-1}) = p(c^{(i)}_{k-1}) = 1$ if i is the root, and $p(c^{(i)}_{k-1}|c^{\text{pa}(i)}_{k-1}) = p(c^{(i)}_{k-1}, c^{\text{pa}(i)}_{k-1})/p(c^{\text{pa}(i)}_{k-1}) = (1)$ otherwise, where (1) should be understood as the matrix form of a conditional probability table.

3.6.2 Predicted target trees

When the filter is already maintaining target trees, predicted target trees must be built to account for dependencies that may have appeared at the current time step. The term *predicted target tree* is proper as the tree is used as input to compute marginal association probabilities and differs from the *posterior target tree* that incorporates posterior component probabilities. The method is presented for a single predicted tree because targets from distinct previous trees[4] that share possible assignments are grouped together, along with their trees[5], into a single cluster, and each cluster is treated separately to originate one predicted tree. For clarity of exposition, we describe the method referring to one previous (posterior) tree, but the procedure is exactly the same in case more than one previous tree become dependent and take part in the same cluster. Given a target tree $G_{T,k-1} = (\mathcal{V}_{T,k-1}, \mathcal{E}_{T,k-1})$ from time step $k-1$, and a set of predicted targets indexed by $\mathcal{I}_{n,k|k-1} = \mathcal{I}_{c,k|k-1} \cup \mathcal{I}_{u,k|k-1}$, a new target tree, $G_{T,k|k-1} = (\mathcal{V}_{T,k|k-1}, \mathcal{E}_{T,k|k-1})$, will be built before applying the EHM 3. The following guidelines are used to build a predicted tree:

- A predicted target tree must keep probabilistic information from previous posterior trees.
- Information from previous trees is considered only for confirmed targets.
- New dependencies among targets of a previous tree, and between targets of the previous tree and other targets are considered through new shared assignments.
- A predicted target tree is a Chow-Liu tree [31], which minimizes the overall loss of information that would follow representing pairs of targets as independent.
- A predicted target tree is constrained to obey the EHM rule, that is, distinct branches have mutually exclusive possible assignments.

[4]Note that a target not pertaining to any previous tree is a single-node tree in itself.

[5]Two distinct target trees are merged in a single cluster if elements of both trees start sharing possible assignments.

We define the set of predicted targets of a previous tree as $\mathcal{I}_{T,k|k-1} \triangleq \mathcal{I}_{n,k|k-1} \cap \mathcal{V}_{T,k-1}$, the set of predicted targets not in the previous tree as $\mathcal{I}_{\bar{T},k|k-1} \triangleq \mathcal{I}_{n,k|k-1} \setminus \mathcal{I}_{T,k|k-1}$, the set of confirmed targets of the previous tree as $\mathcal{I}_{T^{(c)},k|k-1} \triangleq \mathcal{I}_{c,k|k-1} \cap \mathcal{V}_{T,k-1}$, and the set of possible assignments for target i shared only with targets in the previous tree as $\mathcal{M}_{i,T} \triangleq \bigcup_{\ell \in \mathcal{V}_{T,k-1}} \mathcal{M}_\ell \cap \mathcal{M}_i$. We also declare a symmetric matrix of possible relations between distinct targets in the predicted tree as $\mathcal{R}_k : \mathcal{I}_{n,k|k-1} \times \mathcal{I}_{n,k|k-1} \to \{0,1\}$, where $\mathcal{R}_k(i,j) = 1$ means that a pair $(i,j) \in \mathcal{I}_{n,k|k-1} \times \mathcal{I}_{n,k|k-1}$ is probabilistically related, and $\mathcal{R}_k(i,j) = 0$ if (i,j) are not dependent or if $i = j$. Following the guidelines, the relation matrix is a logical composition of three relation matrices as $\mathcal{R}_k = \mathcal{R}_{T^{(c)},k|k-1} \vee \mathcal{R}_{T^{(a)},k} \vee \mathcal{R}_{T-\bar{T},k}$, where $\mathcal{R}_{T^{(c)},k|k-1}$ accounts for probabilistic relations between confirmed targets inherited from the previous tree, $\mathcal{R}_{T^{(a)},k}$ accounts for relations among targets of the previous tree due to new shared assignments, and $\mathcal{R}_{T-\bar{T},k}$ accounts for relations between targets of the previous tree and other targets (not in the previous tree). The algorithm for updating the target tree then follows 6 steps:

1. Construction of relation matrices $\mathcal{R}_{T^{(c)},k|k-1}$, $\mathcal{R}_{T^{(a)},k}$, $\mathcal{R}_{T-\bar{T},k}$, to obtain $\mathcal{R}_k$.
2. Calculation of joint factors, $p(c_{k-1}^{(i),(j)}|\mathrm{Y}_{1:k-1}) \equiv p(c_{k-1}^{(i),(j)}|\mathrm{Y}_{1:k-1})$, for every possibly related pair (i,j) as $\mathcal{R}_k(i,j) = 1$, by marginalizating out components on previous tree(s) or initializing new appearing targets.
3. Prediction of marginal association distributions $p(c_{k-1}^{(i)}, a_k^{(i)}|E_k^{(i)}, \mathrm{Y}_{1:k-1})$ and pairwise joint association distributions $p(c_{k-1}^{(i)}, a_k^{(i)}, c_{k-1}^{(j)}, a_k^{(j)}|E_k^{(i),(j)}, \mathrm{Y}_{1:k-1})$ for every possibly related pair (i,j) as $\mathcal{R}_k(i,j) = 1$.
4. Computation of information loss that would follow representing the joint state distribution of two targets, $p(\mathrm{x}_k^{(i)}, \mathrm{x}_k^{(j)}|E_k^{(i),(j)}, \mathrm{Y}_{1:k-1})$, as if they are independent, for every possibly related pair (i,j) as $\mathcal{R}_k(i,j) = 1$. The information gain of each pair (i,j), arranged in a matrix $\mathcal{D}_k : \mathcal{I}_{n,k|k-1} \times \mathcal{I}_{n,k|k-1} \to \mathbb{R}$, is evaluated as the Kullback–Leibler divergence between the pairwise joint state distribution and the product of marginal single-target distributions for targets i and j.
5. Construction of the predicted target tree $G_{T,k|k-1}$ by the Chow-Liu method [31], based on the information gain matrix $\mathcal{D}_k$, under the EHM constraint.
6. Selection of marginal factors, $p(c_{k-1}^{(i)}|\mathrm{Y}_{1:k-1})$, and recalculation of conditional factors, $p(c_{k-1}^{(i)}|c_{k-1}^{\mathrm{pa}(i)}, \mathrm{Y}_{1:k-1})$, for the predicted tree based on the joint factors calculated in Step 2.

Step 1

The relation matrix $\mathcal{R}_{T^{(c)},k|k-1}$ describes the probabilistic relations between confirmed targets inherited from a previous tree, and is constituted as

$$\mathcal{R}_{T^{(c)},k|k-1}(i,j) = \mathcal{R}_{T^{(c)},k|k-1}(j,i) \triangleq \begin{cases} 1, & i \in \mathrm{ch}(j) \text{ or } j \in \mathrm{ch}(i) \text{ and } i,j \in \mathcal{I}_{T^{(c)},k|k-1}, \\ 0, & \text{otherwise.} \end{cases} \tag{3.78}$$

This matrix can be built by sweeping a previous tree from each leaf to the root, and setting $\mathcal{R}_{T^{(c)},k|k-1}(\mathrm{pa}(i), i) = 1$ every time $i, \mathrm{pa}(i) \in \mathcal{I}_{T^{(c)},k|k-1}$. If more than one previous tree are

in the cluster under analysis, one such matrix $\mathcal{R}^{(\ell)}_{T^{(c)},k|k-1}$ is obtained per previous tree $G^{(\ell)}_{T,k-1}$ and the final matrix is composed as $\mathcal{R}_{T^{(c)},k|k-1} = \bigvee_\ell \mathcal{R}^{(\ell)}_{T^{(c)},k|k-1}$.

The relation matrix $\mathcal{R}_{T^{(a)},k}$ describes relations among targets of a previous tree due to new shared assignments and is composed as

$$\mathcal{R}_{T^{(a)},k}(i,j) = \mathcal{R}_{T^{(a)},k}(j,i) \triangleq \begin{cases} 1, & \mathcal{M}_i \cap \mathcal{M}_j \neq \emptyset \text{ and } i,j \in \mathcal{I}_{T,k|k-1}, \\ 0, & \text{otherwise.} \end{cases} \tag{3.79}$$

Again, if more than one previous tree are in the cluster under analysis, one such matrix $\mathcal{R}^{(\ell)}_{T^{(a)},k}$ is obtained per previous tree $G^{(\ell)}_{T,k-1}$ and the final matrix is composed as $\mathcal{R}_{T^{(a)},k} = \bigvee_\ell \mathcal{R}^{(\ell)}_{T^{(a)},k}$. The relation matrix $\mathcal{R}_{T-\bar{T},k}$ accounts for relations between targets of a previous tree and other targets not in the same tree, and is constituted as

$$\mathcal{R}_{T-\bar{T},k}(i,j) = \mathcal{R}_{T-\bar{T},k}(j,i) \triangleq \begin{cases} 1, & \mathcal{M}_i \cap \mathcal{M}_j \neq \emptyset \text{ and } i \in \mathcal{I}_{T,k|k-1}, j \in \mathcal{I}_{\bar{T},k|k-1}, \\ 0, & \text{otherwise.} \end{cases} \tag{3.80}$$

In fact, given a target $i \in \mathcal{I}_{T,k|k-1}$, a composite matrix $\mathcal{R}_{T^{(a)},T-\bar{T},k}(i,j) \triangleq \mathcal{R}_{T^{(a)},k}(i,j) \vee \mathcal{R}_{T-\bar{T},k}(i,j)$ can be constructed at once by verifying the assignments for a target $j \in \mathcal{I}_{\bar{T},k|k-1}$ that are shared only with targets of the previous tree: if $\mathcal{M}_i \cap \mathcal{M}_{j,T} \neq \emptyset$ then $\mathcal{R}_{T^{(a)},T-\bar{T},k}(i,j) = \mathcal{R}_{T^{(a)},T-\bar{T},k}(j,i) = 1$. The same reasoning regarding multiple previous trees applies for this matrix, where one such matrix $\mathcal{R}^{(\ell)}_{T^{(a)},T-\bar{T},k}$ per previous tree $G^{(\ell)}_{T,k-1}$ can be built and the final matrix is composed as $\mathcal{R}_{T^{(a)},T-\bar{T},k} = \bigvee_\ell \mathcal{R}^{(\ell)}_{T^{(a)},T-\bar{T},k}$. The complete relation matrix is given by $\mathcal{R}_k = \mathcal{R}_{T^{(c)},k|k-1} \vee \mathcal{R}_{T^{(a)},T-\bar{T},k}$.

Step 2

For every possibly related pair $(i,j) \in \mathcal{I}_{n,k|k-1} \times \mathcal{I}_{n,k|k-1}$, as marked by $\mathcal{R}_k(i,j) = 1$, joint factors $p(c^{(i),(j)}_{k-1}|\mathrm{Y}_{1:k-1})$ must be calculated by marginalizing out components on previous tree(s) or initializing new appearing targets. These joint factors are required by Step 3 to compute predicted marginal association distributions, and by Step 5 to attribute probability factors to the predicted tree. For any two targets in a previous tree, $i,j \in G_{T,k-1}$, we can define a path of nodes between them as

$$\mathcal{P}_{]i,j[} \triangleq \{\ell \in \mathcal{V}_{T,k-1} | \ell \in \mathrm{de}(i) \cap \mathrm{an}(j) \text{ or } \ell \in \mathrm{an}(i) \cap \mathrm{de}(j)\}. \tag{3.81}$$

Thus, the joint probability distribution on the extremal nodes of path $\mathcal{P}_{]i,j[}$ is given by

$$p(c^{(i),(j)}_{k-1}|\mathrm{Y}_{1:k-1}) \propto \sum_{c^{\mathcal{P}_{]i,j[}}_{k-1}} \prod_{\ell \in \mathcal{P}_{]i,j[}} p(c^{(\ell)}_{k-1}|c^{\mathrm{pa}(\ell)}_{k-1}, \mathrm{Y}_{1:k-1}),\ i,j \in \mathcal{V}_{T,k-1}. \tag{3.82}$$

If at least one of the targets i or j does not pertain to the previous tree $G_{T,k-1}$, then

$$p(c^{(i),(j)}_{k-1}|\mathrm{Y}_{1:k-1}) \propto p(c^{(i)}_{k-1}|\mathrm{Y}_{1:k-1}) p(c^{(j)}_{k-1}|\mathrm{Y}_{1:k-1}),\ i \text{ or } j \notin \mathcal{V}_{T,k-1}, \tag{3.83}$$

where each marginal factor $p(c^{(i)}_{k-1}|\mathrm{Y}_{1:k-1})$ is retrieved from the structure of a previous tree, a previously isolated node (single-node "tree"), or a newborn target (not in a tree). These

marginal factors are stored with the structure of previous trees, and can be identified as

$$p(c_{k-1}^{(i)}|\mathrm{Y}_{1:k-1}) = \begin{cases} \sum_{c_{k-1}^{\mathcal{V}_{T_s,k-1}} \setminus c_{k-1}^{(i)}} \prod_{\ell \in \mathcal{V}_{T_s,k-1}} p(c_{k-1}^{(\ell)}|c_{k-1}^{\mathrm{pa}(\ell)}, \mathrm{Y}_{1:k-1}), & i \in \mathcal{V}_{T_s,k-1} \text{ (any previous tree } s\text{)}, \\ 1, & i \notin \mathcal{V}_{T_s,k-1} \text{ (none of trees)}. \end{cases} \tag{3.84}$$

If both $i, j \notin \mathcal{V}_{T_s,k-1}$ (none of previous trees), then $p(c_{k-1}^{(i),(j)}|\mathrm{Y}_{1:k-1}) \coloneqq (1)$, where (1) should be understood as the matrix form of a joint probability table. If more than one previous tree are in the cluster under analysis, the same rules apply, where (3.82) is used if two targets pertain to the same previous tree, and (3.83) is used otherwise. Each previously "isolated target", not pertaining to any previous tree, is characterized as a single-node tree with component probability distribution $p(c_{k-1}^{(i)}|\mathrm{Y}_{1:k-1})$.

Step 3

For every possibly related pair (i, j), marginal association distributions $p(c_{k-1}^{(i)}, a_k^{(i)}|E_k^{(i)}, \mathrm{Y}_{1:k-1})$ and pairwise joint association distributions $p(c_{k-1}^{(i),(j)}, a_k^{(i),(j)}|E_k^{(i),(j)}, \mathrm{Y}_{1:k-1})$ must be computed. These association distributions are required by Step 4 to compute the estimated gain of information. The predicted marginal association distributions are computed based on the factors obtained in Step 2 as

$$p(c_{k-1}^{(i)}, a_k^{(i)}|E_k^{(i)}, \mathrm{Y}_{1:k-1}) \propto \alpha^{\mathcal{I}_{\bar{a},k} \cap \{i\}} \frac{p(c_{k-1}^{(i)}|\mathrm{Y}_{1:k-1}) \psi(a_k^{(i)}, E_k^{(i)}|c_{k-1}^{(i)})}{p(E_k^{(i)}|\mathrm{Y}_{1:k-1})}, \tag{3.85}$$

$$p(c_{k-1}^{(i),(j)}, a_k^{(i),(j)}|E_k^{(i),(j)}, \mathrm{Y}_{1:k-1}) \propto \alpha^{\mathcal{I}_{\bar{a},k} \cap \{i,j\}} \times \frac{p(c_{k-1}^{(i),(j)}|\mathrm{Y}_{1:k-1}) \psi(a_k^{(i)}, E_k^{(i)}|c_{k-1}^{(i)}) \psi(a_k^{(j)}, E_k^{(j)}|c_{k-1}^{(j)})}{p(E_k^{(i)}|\mathrm{Y}_{1:k-1}) p(E_k^{(j)}|\mathrm{Y}_{1:k-1})}, \tag{3.86}$$

for all targets $i, j \in \mathcal{I}_{n,k|k-1}$ involved in possibly related pairs (i, j) marked by $\mathcal{R}_k(i, j) = 1$.

Step 4

For every possibly related pair of targets (i, j), marked by $\mathcal{R}_k(i, j) = 1$, the loss of information that would follow representing the pairwise joint state distribution, $p(\mathrm{x}_k^{(i),(j)}|E_k^{(i),(j)}, \mathrm{Y}_{1:k-1})$, as if targets are independent, must be evaluated. The information gain by representing each pair (i, j) as dependent is arranged in a matrix $\mathcal{D}_k : \mathcal{I}_{n,k|k-1} \times \mathcal{I}_{n,k|k-1} \to \mathbb{R}$, where each element is defined as

$$\mathcal{D}_k(i, j) = \mathcal{D}_k(j, i) \triangleq \begin{cases} D_{\mathrm{KL}}(p(\mathrm{x}_k^{(i),(j)}|E_k^{(i),(j)}, \mathrm{Y}_{1:k-1})||q(\mathrm{x}_k^{(i),(j)}|E_k^{(i),(j)}, \mathrm{Y}_{1:k-1})), & \mathcal{R}_k(i, j) = 1, \\ 0, & \text{otherwise}, \end{cases} \tag{3.87}$$

where $D_{\mathrm{KL}}(\mathcal{P}||\mathcal{Q})$ is the Kullback–Leibler divergence of distribution $\mathcal{P}$ with respect to distribution $\mathcal{Q}$, and

$$\begin{aligned} p(\mathrm{x}_k^{(i),(j)}|E_k^{(i),(j)},\mathrm{Y}_{1:k-1}) = &\sum_{a_k^{(i),(j)}}\sum_{c_{k-1}^{(i),(j)}} p(c_{k-1}^{(i),(j)},a_k^{(i),(j)}|E_k^{(i),(j)},\mathrm{Y}_{1:k-1}) \\ &\times p(\mathrm{x}_k^{(i)}|a_k^{(i)},c_{k-1}^{(i)},E_k^{(i)},\mathrm{Y}_{1:k-1})p(\mathrm{x}_k^{(j)}|a_k^{(j)},c_{k-1}^{(j)},E_k^{(j)},\mathrm{Y}_{1:k-1}), \quad (3.88) \\ q(\mathrm{x}_k^{(i),(j)}|E_k^{(i),(j)},\mathrm{Y}_{1:k-1}) = &\sum_{a_k^{(i),(j)}}\sum_{c_{k-1}^{(i),(j)}} p(c_{k-1}^{(i)},a_k^{(i)}|E_k^{(i)},\mathrm{Y}_{1:k-1})p(c_{k-1}^{(j)},a_k^{(j)}|E_k^{(j)},\mathrm{Y}_{1:k-1}) \\ &\times p(\mathrm{x}_k^{(i)}|a_k^{(i)},c_{k-1}^{(i)},E_k^{(i)},\mathrm{Y}_{1:k-1})p(\mathrm{x}_k^{(j)}|a_k^{(j)},c_{k-1}^{(j)},E_k^{(j)},\mathrm{Y}_{1:k-1}). \quad (3.89) \end{aligned}$$

Note that the Kullback–Leibler divergence is a measure of the amount of information lost when q (independent targets) is used to approximate p (dependent targets) in (3.87). Thus, the most informative tree maximizes $\mathcal{D}_k$ over the connected nodes (minimizes the information loss). Since p and q are mixtures possibly involving many terms, an approximated calculation of the KL-divergence must be used in favor of computational efficiency. We advocate the method proposed in [81] to compute KL-divergences between mixtures by means of the bound

$$D_{\mathrm{KL}}\left(\sum_{\ell=1}^{M}\alpha_\ell f_\ell(\mathrm{x})\,\middle\|\,\sum_{\ell=1}^{M}\beta_\ell g_\ell(\mathrm{x})\right) \leq \sum_{\ell=1}^{M}\alpha_\ell D_{\mathrm{KL}}(f_\ell(\mathrm{x})||g_\ell(\mathrm{x})) + \sum_{\ell=1}^{M}\alpha_\ell \log\frac{\alpha_\ell}{\beta_\ell}. \quad (3.90)$$

In our case $f_\ell(\mathrm{x}_k^{(i),(j)}) = g_\ell(\mathrm{x}_k^{(i),(j)}) = p(\mathrm{x}_k^{(i)}|a_k^{(i)},c_{k-1}^{(i)},E_k^{(i)},\mathrm{Y}_{1:k-1})p(\mathrm{x}_k^{(j)}|a_k^{(j)},c_{k-1}^{(j)},E_k^{(j)},\mathrm{Y}_{1:k-1})$, which results $D_{\mathrm{KL}}(f_\ell||g_\ell) = 0$. Therefore, we approximate (3.87) as

$$\begin{aligned} &D_{\mathrm{KL}}(p(\mathrm{x}_k^{(i),(j)}|E_k^{(i),(j)},\mathrm{Y}_{1:k-1})||q(\mathrm{x}_k^{(i),(j)}|E_k^{(i),(j)},\mathrm{Y}_{1:k-1})) \\ &\qquad\approx \sum_{a_k^{(i),(j)}}\sum_{c_{k-1}^{(i),(j)}} p(c_{k-1}^{(i),(j)},a_k^{(i),(j)}|E_k^{(i),(j)},\mathrm{Y}_{1:k-1}) \\ &\qquad\quad\times\log\left(\frac{p(c_{k-1}^{(i),(j)},a_k^{(i),(j)}|E_k^{(i),(j)},\mathrm{Y}_{1:k-1})}{p(c_{k-1}^{(i)},a_k^{(i)}|E_k^{(i)},\mathrm{Y}_{1:k-1})p(c_{k-1}^{(j)},a_k^{(j)}|E_k^{(j)},\mathrm{Y}_{1:k-1})}\right), \quad (3.91) \end{aligned}$$

which is very fast to compute.

STEP 5

Based on the information gain matrix, $\mathcal{D}_k$, the predicted target tree $G_{T,k|k-1}$ is built by the Chow-Liu algorithm [31], except that choices of connections between nodes are constrained by the EHM structure, that is, targets in distinct branches have mutually exclusive sets of possible assignments. The relation matrix $\mathcal{R}_k$ indicates which targets in the new tree could be connected. The following steps are taken to build the Chow-Liu tree:

1. The root is started at the root of (one of) the previous tree(s), $\mathrm{rt} = \mathrm{rt}_{T,k-1}$, with $\mathrm{pa}(\mathrm{rt}) \coloneqq \emptyset$. Let $\mathcal{C}_{r,k} \triangleq \{s \in \mathcal{I}_{n,k|k-1} | \mathcal{R}_k(r,s) = 1\}$ be the set of all targets that can be connected to a target r. A target $i_1 \in \mathcal{C}_{\mathrm{rt},k}$ satisfying $i_1 = \mathrm{argmax}_{\ell\in\mathcal{C}_{\mathrm{rt},k}}\mathcal{D}_k(\mathrm{rt},\ell)$ such that $\mathcal{M}_{i_1}\cap\mathcal{M}_{\mathrm{rt}} \neq \emptyset$ is chosen as the first child, i.e., $\mathrm{ch}(\mathrm{rt}) \coloneqq \{i_1\}$, $\mathrm{pa}(i_1) \coloneqq \{\mathrm{rt}\}$, $\mathrm{ch}(i_1) \coloneqq \emptyset$, and the current set of nodes is updated as $\mathcal{V}_{T,k|k-1} = \{\mathrm{rt}, i_1\}$.

2. Next, each node already added in the tree is verified in turn. For each node $r \in$

$\mathcal{V}_{T,k|k-1}$, the potential children $\mathcal{C}_{r,k} \setminus \mathrm{ch}(r)$ are verified. If a target s satisfying $s = \mathrm{argmax}_{\ell \in \mathcal{C}_{r,k} \setminus \mathrm{ch}(r)} \mathcal{D}_k(r, \ell)$ such that $\mathcal{M}_s \cap \mathcal{M}_r \neq \emptyset$ and $\mathcal{M}_s \cap \mathcal{M}_{\mathrm{ch}(r)} = \emptyset$ is found, then it is kept as a tentative child of r, i.e. $\breve{\mathrm{pa}}(s) = \{r\}$ ($\breve{\mathrm{pa}}$ denoting "potential parent"). Once all nodes in $\mathcal{V}_{T,k|k-1}$ were iterated and each has a tentative child, the child that would give the maximum information gain, that is, $s = \mathrm{argmax}_{r \in \mathcal{V}_{T,k|k-1}, \ell \in \mathcal{C}_{r,k} \setminus \mathrm{ch}(r)} \mathcal{D}_k(r, \ell)$, is added as $\mathrm{ch}(\breve{\mathrm{pa}}(s)) \leftarrow \mathrm{ch}(\breve{\mathrm{pa}}(s)) \cup \{s\}$, $\mathrm{pa}(s) := \breve{\mathrm{pa}}(s)$, $\mathrm{ch}(s) := \emptyset$, and the current set of nodes is updated as $\mathcal{V}_{T,k|k-1} \leftarrow \mathcal{V}_{T,k|k-1} \cup \{s\}$.

3. When all targets in the cluster have been used in the new tree[6], that is, $\mathcal{V}_{T,k|k-1} = \mathcal{I}_{n,k|k-1}$, the tree is complete.

Step 6

The marginal factors $p(c_{k-1}^{(i)}|\mathrm{Y}_{1:k-1})$ are selected (from previous trees) for $i \in G_{T,k|k-1}$, and the conditional factors $p(c_{k-1}^{(i)}|c_{k-1}^{\mathrm{pa}(i)}, \mathrm{Y}_{1:k-1})$ are recalculated for the predicted tree based on the joint factors computed in Step 2. The conditional component distributions are evaluated by

$$p(c_{k-1}^{(i)}|c_{k-1}^{\mathrm{pa}(i)}, \mathrm{Y}_{1:k-1}) = \frac{p(c_{k-1}^{(i),\mathrm{pa}(i)}|\mathrm{Y}_{1:k-1})}{p(c_{k-1}^{\mathrm{pa}(i)}|\mathrm{Y}_{1:k-1})}, \text{ s.t. } i, \mathrm{pa}(i) \in G_{T,k|k-1}. \tag{3.92}$$

The selected and recomputed factors, $p(c_{k-1}^{(i)}|\mathrm{Y}_{1:k-1})$ and $p(c_{k-1}^{(i)}|c_{k-1}^{\mathrm{pa}(i)}, \mathrm{Y}_{1:k-1})$, are attributed to the predicted tree.

3.6.3 Update of target trees

Based on posterior association probabilities computed via EHM 3, $p(c_{k-1}^{(i)}, a_k^{(i)}|E_k^{(i)}, \mathrm{Y}_{1:k})$ and $p(c_{k-1}^{(i),\mathrm{pa}(i)}, a_k^{(i),\mathrm{pa}(i)}|E_k^{(i),\mathrm{pa}(i)}, \mathrm{Y}_{1:k})$, factors on the predicted target trees must be expressed in terms of posterior component distributions, $p(c_k^{(i)}|\mathrm{Y}_{1:k})$ and $p(c_k^{(i)}|c_k^{\mathrm{pa}(i)}, \mathrm{Y}_{1:k})$. Observe that

$$c_k^{(i)} \triangleq (c_{k-1}^{(i)}, a_k^{(i)})|E_k^{(i)}, \tag{3.93}$$

that is, conditioned on target existence, the composition of each previous component $c_{k-1}^{(i)} \in \Gamma_{k-1}^{(i)}$ of the ith target with every possible assignment $a_k^{(i)} \in \mathcal{M}_i \cup \{0\}$ gives origin to a posterior component $c_k^{(i)} \in \Gamma_k^{(i)} = \Gamma_{k-1}^{(i)} \times \mathcal{M}_i \cup \{0\}$. Thus,

$$p(c_k^{(i)}|\mathrm{Y}_{1:k}) \equiv p(c_{k-1}^{(i)}, a_k^{(i)}|E_k^{(i)}, \mathrm{Y}_{1:k}), \tag{3.94}$$

$$p(c_k^{(i)}|c_k^{\mathrm{pa}(i)}, \mathrm{Y}_{1:k}) = \frac{p(c_k^{(i),\mathrm{pa}(i)}|\mathrm{Y}_{1:k})}{p(c_k^{\mathrm{pa}(i)}|\mathrm{Y}_{1:k})} \equiv \frac{p(c_{k-1}^{(i),\mathrm{pa}(i)}, a_k^{(i),\mathrm{pa}(i)}|E_k^{(i),\mathrm{pa}(i)}, \mathrm{Y}_{1:k})}{p(c_{k-1}^{\mathrm{pa}(i)}, a_k^{\mathrm{pa}(i)}|E_k^{\mathrm{pa}(i)}, \mathrm{Y}_{1:k})}. \tag{3.95}$$

3.7 Mixture reduction and track management

3.7.1 Mixture reduction

As is the case of any multi-target filter with target state densities based on mixtures, the JPDA-EHM3 requires some mechanism to limit growth of the component number. Since for every

[6] Recall that the description was made for a single cluster to simplify exposition. If more than one target cluster exist, then one target tree $G_{T,k}^{(g)}$ per cluster $g \in \mathbb{N}$ must be constructed.

mixture component several association hypotheses are possible, the number of components per target grows exponentially in time if more than one measurement is validated per component. The usual approach to keep a reasonable number of components is proposed in [153], consisting of merging components in the mixture that are similar in some sense, while presetting a desired number of components for each filtering cycle. We advocate merging components that have been created with the same measurement association, i.e.,

$$\begin{aligned} p(\mathrm{x}_k^{(i)}|E_k^{(i)},\mathrm{Y}_{1:k}) &= \sum_{a_k^{(i)}\in\mathcal{M}_i\cup\{0\}}\sum_{c_{k-1}^{(i)}\in\Gamma_{k-1}^{(i)}} p(\mathrm{x}_k^{(i)}|c_{k-1}^{(i)},a_k^{(i)},E_k^{(i)},\mathrm{Y}_{1:k})p(c_{k-1}^{(i)},a_k^{(i)}|E_k^{(i)},\mathrm{Y}_{1:k}) \\ &= \sum_{a_k^{(i)}\in\mathcal{M}_i\cup\{0\}} p(\mathrm{x}_k^{(i)}|a_k^{(i)},E_k^{(i)},\mathrm{Y}_{1:k})p(a_k^{(i)}|E_k^{(i)},\mathrm{Y}_{1:k}) \\ &\equiv \sum_{c_k^{(i)}\in\Gamma_k^{(i)}} p(\mathrm{x}_k^{(i)}|c_k^{(i)},E_k^{(i)},\mathrm{Y}_{1:k})p(c_k^{(i)}|\mathrm{Y}_{1:k}), \end{aligned} \tag{3.96}$$

such that $c_k^{(i)}\equiv a_k^{(i)}|E_k^{(i)}$ and so $\Gamma_k^{(i)}\equiv\mathcal{M}_i\cup\{0\}$. Assuming Gaussian components

$$p(\mathrm{x}_k^{(i)}|c_{k-1}^{(i)},a_k^{(i)},E_k^{(i)},\mathrm{Y}_{1:k}) = \mathcal{N}(\mathrm{x}_k^{(i)}|\mu_{c_{k-1}^{(i)},a_k^{(i)}},\Sigma_{c_{k-1}^{(i)},a_k^{(i)}}), \tag{3.97}$$

$$p(c_{k-1}^{(i)},a_k^{(i)}|E_k^{(i)},\mathrm{Y}_{1:k}) = w_{c_{k-1}^{(i)},a_k^{(i)}}, \tag{3.98}$$

the mixture reduction obtains $p(c_k^{(i)}|\mathrm{Y}_{1:k}) = w_{c_k^{(i)}}$ and $p(\mathrm{x}_k^{(i)}|c_k^{(i)},E_k^{(i)},\mathrm{Y}_{1:k}) = \mathcal{N}(\mathrm{x}_k^{(i)}|\mu_{c_k^{(i)}},\Sigma_{c_k^{(i)}})$ as

$$w_{c_k^{(i)}} = \sum_{c_{k-1}^{(i)}\in\Gamma_{k-1}^{(i)}} w_{c_{k-1}^{(i)},a_k^{(i)}}, \tag{3.99}$$

$$\mu_{c_k^{(i)}} = \frac{1}{w_{c_k^{(i)}}}\sum_{c_{k-1}^{(i)}\in\Gamma_{k-1}^{(i)}} w_{c_{k-1}^{(i)},a_k^{(i)}}\mu_{c_{k-1}^{(i)},a_k^{(i)}}, \tag{3.100}$$

$$\begin{aligned} \Sigma_{c_k^{(i)}} &= \frac{1}{w_{c_k^{(i)}}}\sum_{c_{k-1}^{(i)}\in\Gamma_{k-1}^{(i)}} w_{c_{k-1}^{(i)},a_k^{(i)}}\Sigma_{c_{k-1}^{(i)},a_k^{(i)}} \\ &\quad + \frac{1}{w_{c_k^{(i)}}}\sum_{c_{k-1}^{(i)}\in\Gamma_{k-1}^{(i)}} \left(\mu_{c_{k-1}^{(i)},a_k^{(i)}}-\mu_{c_k^{(i)}}\right)\left(\mu_{c_{k-1}^{(i)},a_k^{(i)}}-\mu_{c_k^{(i)}}\right)^T. \end{aligned} \tag{3.101}$$

Thus, all information from previous components is consolidated in the new components on a per association basis. The advantage of this scheme is maintaining only one component for the null hypothesis, $a_k^{(i)}=0$, which is usually uninformative, and making the total number of components minimally sufficient to explain all association hypotheses (conditioned on existence). In this setting, the number of mixture components does not grow exponentially with time, but is rather determined by association possibilities and so scales with the scenario complexity.

Once the mixture reduction is applied, the posterior target tree must be updated to account

for the modification in the number of components. The posterior target tree is updated as

$$\begin{aligned} p(c_k^{(i)}|\mathrm{Y}_{1:k}) &\equiv p(a_k^{(i)}|E_k^{(i)},\mathrm{Y}_{1:k}) \\ &= \sum_{c_{k-1}^{(i)}\in\Gamma_{k-1}^{(i)}} p(c_{k-1}^{(i)},a_k^{(i)}|E_k^{(i)},\mathrm{Y}_{1:k}) = w_{c_k^{(i)}}, \end{aligned} \tag{3.102}$$

$$\begin{aligned} p(c_k^{(i)}|c_k^{\mathrm{pa}(i)},\mathrm{Y}_{1:k}) &\equiv p(a_k^{(i)}|a_k^{\mathrm{pa}(i)},E_k^{(i),\mathrm{pa}(i)},\mathrm{Y}_{1:k}) \\ &= \frac{p(a_k^{(i),\mathrm{pa}(i)}|E_k^{(i),\mathrm{pa}(i)},\mathrm{Y}_{1:k})}{p(a_k^{\mathrm{pa}(i)}|E_k^{\mathrm{pa}(i)},\mathrm{Y}_{1:k})} \\ &= \frac{\sum_{c_{k-1}^{(i)}\in\Gamma_{k-1}^{(i)}}\sum_{c_{k-1}^{\mathrm{pa}(i)}\in\Gamma_{k-1}^{\mathrm{pa}(i)}} p(c_{k-1}^{(i),\mathrm{pa}(i)},a_k^{(i),\mathrm{pa}(i)}|E_k^{(i),\mathrm{pa}(i)},\mathrm{Y}_{1:k})}{\sum_{c_{k-1}^{\mathrm{pa}(i)}\in\Gamma_{k-1}^{\mathrm{pa}(i)}} p(c_{k-1}^{\mathrm{pa}(i)},a_k^{\mathrm{pa}(i)}|E_k^{\mathrm{pa}(i)},\mathrm{Y}_{1:k})} \\ &= \frac{\sum_{c_{k-1}^{(i)}\in\Gamma_{k-1}^{(i)}}\sum_{c_{k-1}^{\mathrm{pa}(i)}\in\Gamma_{k-1}^{\mathrm{pa}(i)}} p(c_{k-1}^{(i),\mathrm{pa}(i)},a_k^{(i),\mathrm{pa}(i)}|E_k^{(i),\mathrm{pa}(i)},\mathrm{Y}_{1:k})}{w_{c_k^{\mathrm{pa}(i)}}}. \end{aligned} \tag{3.103}$$

3.7.2 Track management

The track management is performed based on *Assumption 3.8.* The JPDA-EHM3 provides a framework for keeping $n_{u,k}$ unconfirmed targets, indexed by $\mathcal{I}_{u,k}$, which remain unconfirmed if their posterior probability of existence is within the interval $p(E_k^{(i)}|\mathrm{Y}_{1:k}) \in (\tau_{del}, \tau_{\mathrm{conf}})$, where τ_{del} is a lower threshold for track deletion and τ_{conf} is an upper threshold for track confirmation. At the end of each filtering cycle, if an unconfirmed target is estimated with $p(E_k^{(i)}|\mathrm{Y}_{1:k}) > \tau_{\mathrm{conf}}$, then the corresponding track is confirmed, removed from the unconfirmed set, and incorporated in a set of $n_{c,k}$ confirmed targets, indexed by $\mathcal{I}_{c,k}$. In addition, if any confirmed or unconfirmed target has $p(E_k^{(i)}|\mathrm{Y}_{1:k}) < \tau_{\mathrm{del}}$, then its track is deleted. The practitioner does not have to worry about reflecting any changes in the posterior target tree while performing track management because in the next filtering cycle the algorithm for generating the predicted target tree will automatically incorporate changes about targets that have been confirmed or deleted.

3.8 Numerical experiment

A scenario has been devised to illustrate the JPDA-EHM3 capabilities. A number of targets, unknown a priori, appear in the scene. At first, pairs of targets are set to move into mutual proximity and stay in this condition for a couple of time steps. Then, the targets get separated while moving into mutual proximity with other targets constituting new pairs. The new pairs stay close for another couple of time steps and the scenario is ended by separating all targets. The ground truth for such scenario can be seen in Figure 3.4.

The surveillance region covers the rectangular area $[-2000, +2000]\times[-1500, +1500]$ (m$\times$m), on which two sensors, S1 and S2, collect observations possibly affected by false alarms and corrupted by noise. Both sensors are assumed with a single resolution cell covering the entire surveillance region, with the same measurement accuracy. Sensor S1 collects only position measurements in the Cartesian space at a high rate, one scan per second, and sensor S2 collects (Cartesian) position measurements along with observations of radar cross sections, but at a fairly slow rate, one scan per 70 seconds. The scenario cover $T = 160\,\mathrm{s}$, and so sensor S1 provides 160 collections of position measurements (starting at $t = 1\,\mathrm{s}$) and sensor S2 provides

only two collections of measurements (at $t = 71\,\mathrm{s}$ and $t = 141\,\mathrm{s}$) including target position and cross-section. The scenario is complicated as one or more observations provided by sensor S2 may miss the radar cross-section measurement. This example has been designed to force association ambiguity, via pairwise target proximity, and provide incomplete information[7] for enabling the analyzed filters to solve the ambiguities. The intent is to show that, in these type of scenarios involving association ambiguity and partial observability, maintaining the dependency between targets is crucial.

Each target is described by its state vector $\mathrm{x} = (p_x, p_y, v_x, v_y, \alpha)^T$, where $\mathrm{p} = (p_x, p_y)^T$ is a pair that specifies a position in Cartesian coordinates, $\mathrm{v} = (v_x, v_y)^T$ is the pair specifying velocity in the same coordinates, and α is the target radar cross-section (m^2). At each time instant k, the state is written as $\mathrm{x}_k = \mathrm{x}(t_k)$. For filtering purposes, each target is assumed to move with nearly-constant velocity and its radar cross-section is assumed to evolve as a random-walk process, with transition matrix and state process covariance matrix given respectively by

$$\mathrm{F} = \begin{pmatrix} \mathbb{I}_2 & \mathbb{I}_2\Delta t & 0_{2\times 1} \\ 0_{2\times 2} & \mathbb{I}_2 & 0_{2\times 1} \\ 0_{1\times 2} & 0_{1\times 2} & 1 \end{pmatrix}, \; \mathrm{Q} = \begin{pmatrix} \mathbb{I}_2\sigma_{\delta v}^2\Delta t^3/3 & \mathbb{I}_2\sigma_{\delta v}^2\Delta t^2/2 & 0_{2\times 1} \\ \mathbb{I}_2\sigma_{\delta v}^2\Delta t^2/2 & \mathbb{I}_2\sigma_{\delta v}^2\Delta t & 0_{2\times 1} \\ 0_{1\times 2} & 0_{1\times 2} & \sigma_{\delta\alpha}^2\Delta t \end{pmatrix}, \tag{3.104}$$

where $\mathbb{I}_2$ is the 2×2 identity matrix, $0_{d_1\times d_2}$ is a $d_1 \times d_2$ zero matrix, $\Delta t = 1\,\mathrm{s}$ is the sampling period, and the standard deviation of velocity increments and radar cross-section increments are given respectively by $\sigma_{\delta v} = 2\,\mathrm{m/s/s}^{\frac{1}{2}}$ and $\sigma_{\delta\alpha} = 0.001\,\mathrm{m}^2/\mathrm{s}^{\frac{1}{2}}$. Sensors S1 and S2 collect measurements of position in the Cartesian space (and radar cross-section for S2), corrupted by a Gaussian-distributed noise, characterized by the output matrices and measurement noise covariance matrices as

$$\mathrm{H}_{S1} = \begin{pmatrix} \mathbb{I}_2 & 0_{2\times 3} \end{pmatrix}, \tag{3.105}$$

$$\mathrm{R}_{S1} = \mathrm{R}_p = \begin{pmatrix} \sigma_{p,x}^2 & 0 \\ 0 & \sigma_{p,y}^2 \end{pmatrix}, \tag{3.106}$$

$$\mathrm{H}_{S2} = \begin{pmatrix} \mathbb{I}_2 & 0_{2\times 2} & 0 \\ 0_{1\times 2} & 0_{1\times 2} & 1 \end{pmatrix}, \tag{3.107}$$

$$\mathrm{R}_{S2} = \begin{pmatrix} \mathrm{R}_p & 0_{2\times 2} \\ 0_{1\times 2} & \mathrm{R}_\alpha \end{pmatrix} = \begin{pmatrix} \mathrm{R}_p & 0_{2\times 2} \\ 0_{1\times 2} & \sigma_\alpha^2 \end{pmatrix}, \tag{3.108}$$

where $\sigma_{p,x} = 10\,\mathrm{m}$ and $\sigma_{p,y} = 30\,\mathrm{m}$ are the standard deviations of the measured positions in coordinates x and y respectively, and $\sigma_\alpha = 1\,\mathrm{m}^2$ is the standard deviation of the measured radar cross-section. Each target can be detected with probability $p_d = 0.90$ and measurements are gated with gate-size probability of $p_g = 0.95$. False alarms are generated in Poisson-distributed numbers, with expected rate of $\lambda_{\mathrm{fa}} = 10$ false alarms per second, and the position of each false alarm is sampled uniformly in the surveillance region with "volume" (area) $V = 4000 \times 3000\,\mathrm{m}^2$. The expected volumetric density of false alarms per time frame is given by $\lambda_{\mathrm{fa},V} = \lambda_{\mathrm{fa}}/V$.

Each target remains in the scene up to the next time step with probability $p_s = 0.99$. Newborn targets are dynamically modeled as appearing with rate $\lambda_{b,k|k-1} = \bar{m}_{k-1}/T$ where $\bar{m}_{k-1}$ is the total number of measurements that have not been gated for any of the existing

[7] Radar cross-section provided only twice, and possibly missing for some targets.

targets at step $k-1$, and $T = 160\,\mathrm{s}$. The spatial birth distribution of each newborn target, indexed as $j \in \mathcal{I}_{\bar{m},k-1}$, is given by

$$p_{\mathrm{birth},k|k-1}(\mathrm{x}_k^{(j)}|\mathrm{y}_{k-1}^{\mathcal{I}_{m,k-1}}) = \mathcal{N}(\mathrm{x}_k^{(j)}|\mu_{b,k|k-1}^{(j)}, \Sigma_{b,k|k-1}^{(j)}), \quad j \in \mathcal{I}_{\bar{m},k-1}, \tag{3.109}$$

with probability of existence started as $p(E_k^{(j)}|\mathrm{y}_{k-1}^{(1:m)}) \coloneqq p_{b,k|k-1}^{(j)} = \lambda_{b,k|k-1}$. The mean vector and covariance matrix of each newborn target is given by

$$\mu_{b,k|k-1}^{(j)} = \begin{pmatrix} \mathrm{y}_{p,k-1}^{(j)} \\ 0_{2\times 1} \\ \mathrm{y}_{\alpha,k-1}^{(j)} \end{pmatrix}, \; \Sigma_{b,k|k-1}^{(j)} = \begin{pmatrix} \mathrm{R}_p & 0_{1\times 2} & 0_{2\times 1} \\ 0_{2\times 1} & \mathbb{I}_2 v_{\max}^2/4 & 0_{2\times 1} \\ 0_{1\times 2} & 0_{1\times 2} & \mathrm{R}_\alpha \end{pmatrix}, \tag{3.110}$$

where $\mathrm{y}_{p,k-1}^{(j)}$ and $\mathrm{y}_{\alpha,k-1}^{(j)}$ are the measured position and radar cross-section in the observation $\mathrm{y}_{k-1}^{(j)} \in \mathrm{y}_{k-1}^{\mathcal{I}_{\bar{m},k-1}}$, R_p and R_α are submatrices of the observation noise covariance matrix corresponding to position and radar cross-section respectively, and $v_{\max}$ is the maximum (possible) velocity magnitude in the scenario, set as $v_{\max} \coloneqq D/T$, where $D = \sqrt{4000^2 + 3000^2}$ (m) is the diagonal of the surveillance rectangle. For track management purposes, the lower threshold for track deletion is set as $\tau_{\mathrm{del}} = 0.10$ and the upper threshold for track confirmation $\tau_{\mathrm{conf}} = 0.95$.

We compare the performance of the JPDA-EHM3 with

- a Cardinalized Probability Hypothesis Density (CPHD) filter [129, 203],
- a JIPDA whose associations are resolved by the Global Nearest Neighbor data association algorithm, designated as JIPDA-GNN,
- a JIPDA whose marginal association probabilities are computed by the EHM1, designated as JIPDA-EHM1,
- a JIPDA implemented with one Gaussian mixture per target whose marginal association probabilities are computed by the EHM1 [135], designated as JIPDA-GM-EHM1.

More sophisticated filters such as the Generalized Labeled Multi-Bernoulli filter [164] and DISP [58] have not been included in the comparison because we believe these filters are more comparable to the MHT in terms of required computational resources and performance. The same techniques used for incorporating newborn targets, mixture reduction and track management developed for the JPDA-EHM3 are used for the JIPDA-GNN, JIPDA-EHM1 and JIPDA-GM-EHM1. Note that the JIPDA filter is seen to be equivalent to the multi-Bernoulli filter, a fact that has been studied in the literature [202, 206, 188]. In this sense, our implementation of the JIPDA-GM-EHM1 is very similar to the marginal multi-Bernoulli filter with Poisson birth process presented in [206]. The use of EHM1 for the JIPDA can be justified as the run times based on the classical approach (hypotheses enumeration) for computing marginal association probabilities are prohibitively large. In preliminary tests of our numerical experiment, the JIPDA and JIPDA-GM based on the classical approach performed ten times slower.

The birth process for the CPHD filter has an intensity function modeled by a Gaussian mixture with 10 components, located around the initial position of each target to enable the filter to infer intensity components that explain all targets within the first 15 time steps. For $k \leq 15$, the birth components for the CPHD filter contain the correct cross-section information corresponding to the region where each target is expected. For $k > 15$, these

birth components have radar cross sections sampled from the uniform distribution $\mathcal{U}([5, 1000])$ (m^2). Thus, the birth components are set as $\mu^{(i)}_{b,k|k-1} \coloneqq (\tilde{p}^{(i)}_x, \tilde{p}^{(i)}_y, 0, 0, \tilde{\alpha}^{(i)}_b)^T$ and $\Sigma^{(i)}_{b,k|k-1} \coloneqq \mathrm{diag}(9\sigma^2_{p,x}, 9\sigma^2_{p,y}, v^2_{\max}/4, v^2_{\max}/4, \sigma^2_{\alpha,b})$, where $\tilde{p}^{(i)}_x \sim \mathcal{N}(\cdot|p^{(i)}_x(0), \sigma^2_{p,x})$, $\tilde{p}^{(i)}_y \sim \mathcal{N}(\cdot|p^{(i)}_y(0), \sigma^2_{p,y})$, $\tilde{\alpha}^{(i)}_b \sim \mathcal{N}(\cdot|\alpha^{(i)}(0), \sigma^2_\alpha)$ and $\sigma_{\alpha,b} \coloneqq \sigma_\alpha$ for $k \leq 15$, and $\tilde{\alpha}^{(i)}_b \sim \mathcal{U}([5, 1000])$ and $\sigma_{\alpha,b} \coloneqq (1000 - 5)/2$ otherwise. The maximum number of Gaussian components maintained for the CPHD intensity is 40, with pruning threshold 10^{-5} and merging radius of 4 m. The cardinality distribution estimated by the CPHD is truncated at $n_{\max} = 20$.

In the numerical experiment, 200 Monte Carlo (MC) runs are performed, each with an independently sampled set of target-generated measurements, and independently generated clutter. For all filters, performance is evaluated in terms of:

- mean Optimal Subpattern Assignment (OSPA) metric [174] for cut-off $c_{\mathrm{OSPA}} = 200$ and norm order $p_{\mathrm{OSPA}} = 2$ (including radar cross-section),
- average number of confirmed targets,
- average number of track swaps, and
- computation time (per time step).

3.8.1 Results

Figure 3.4 shows the confirmed tracks generated by the JPDA-EHM3 for an exemplary run. In this run, we observe that once pairs of targets get into mutual proximity, around time step $k = 35$, they become dependent by sharing measurements that are possible to have originated from either target in each pair. As they become dependent, a tree of targets describing their dependency is properly constructed and propagated forward as illustrated in Figure 3.5. These pairwise dependencies are also clear from Figure 3.4 as estimates of either target in each pair, whose uncertainty is represented by consolidated covariance ellipses[8], make evident the association ambiguity, causing the consolidated covariances to occupy a wide shared uncertainty area. Once a pairwise dependency is identified by the algorithm, it is maintained over the subsequent steps, deferring the decision to solve the association ambiguity to future steps. A partially observed set of measurements containing radar cross sections, from sensor S2, helps to solve the association ambiguity at time step $k = 71$, while still keeping the dependency structure. This dependency structure is carried over to step $k = 105$ when other dependencies are identified because new pairs of targets become dependent. At this point, the resulting dependency structure involves all targets, by a transition of pairwise dependencies across all targets. The resulting dependency across all confirmed targets can be seen in Figure 3.6. At time step $k = 141$, additional ambiguities are solved by new measurements from sensor S2, and then the confirmed tracks follow the separated targets without swaps to the end. Note in Figures 3.5 and 3.6 that the previous target trees are from the previous filtering step and may contain unconfirmed targets that may have been deleted via track management. In addition, newborn unconfirmed targets may appear in the predicted trees. In Figures 3.5 and 3.6 labels of unconfirmed targets are grayed out.

[8] The consolidated covariance matrices are obtained by combining the covariances of all mixture components of each target state marginal distribution.

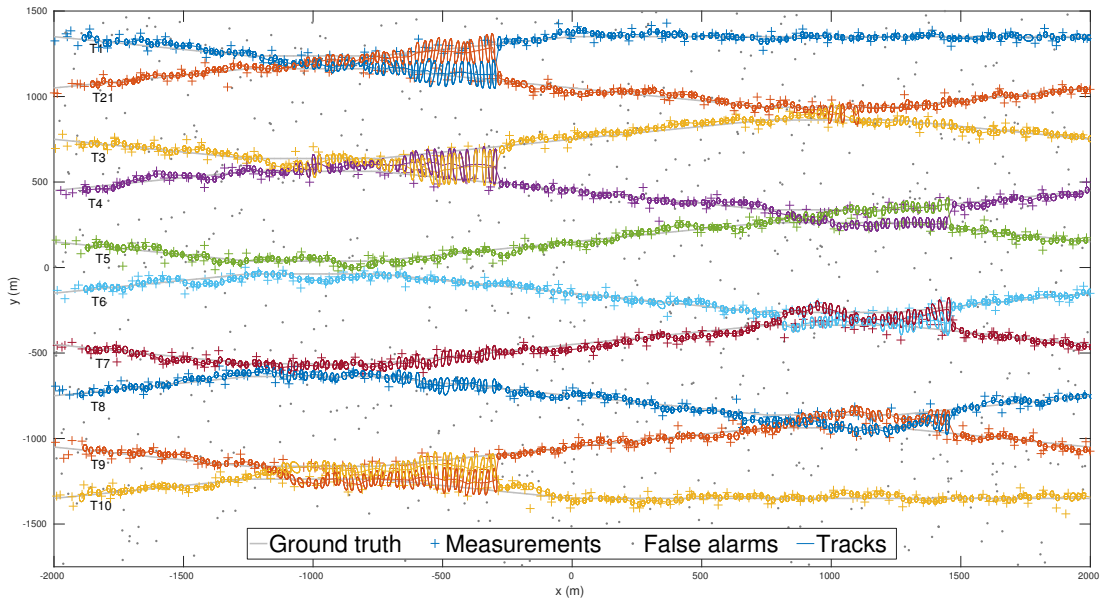


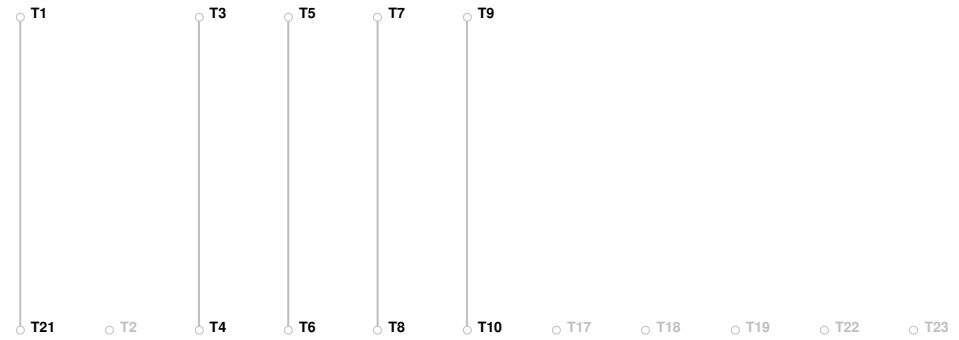
Figure 3.4: Exemplary run: confirmed tracks estimated by JPDA-EHM3.

The performance indexes comparing the JPDA-EHM3, CPHD, JIPDA-GNN, JIPDA-EHM1, and JIPDA-GM-EHM1 can be seen in Figures 3.7–3.10. In terms of confirmed tracks, from Figure 3.7 we can see that all filters perform satisfactorily, except for the JIPDA-GNN, proving the effectiveness of the approach proposed in this chapter to incorporate target birth and track management for confirming targets with high probability of existence. The JIPDA-GNN solves the associations by the global nearest neighbor method, which renders some tracks with a low probability of existence when the wrong association is forced by the association ambiguities.

Regarding track swaps, Figure 3.8 highlights the clear superiority of JPDA-EHM3 owing to its explicit estimation of dependencies between targets, which allows the algorithm to defer hard decisions to future steps when sensible information becomes available. The number of track swaps for the JPDA-EHM3 slightly increases right after close targets start separating because the algorithm is not able to estimate the correct association yet. When information from sensor S2 arrives, partially observed radar cross-section enables disambiguation, such that the JPDA-EHM3 is able to reform its estimates based on the maintained dependency structure. The same does not happen for the JIPDA-GNN, JIPDA-EHM1, and JIPDA-GM-EHM1, which cannot revise their estimates once they are produced. Note that track swaps happen all the time for the CPHD since this filter is not designed to keep track identities.

The track swaps are not identically zero for the JPDA-EHM3 because its ability on revising its estimates depend on the disambiguating information, which may not be available at the critical moment or should be provided for a greater number of time steps to sustain evidence enough for the required inference changes. One can also observe that the JIPDA-GM-EHM1 is slightly better than the JIPDA-EHM1 due to the several mixture components per target in the former, which cater for multiple hypotheses over time. The JIPDA-EHM1 and JIPDA-GM-EHM1 are much better than the JIPDA-GNN in terms of track swaps because the later takes a greedy approach to solve association ambiguities.

As a partial consequence of the performances observed for the track swaps, the mean OSPA metric presented for the JPDA-EHM3 is much better than those presented for the JIPDA-GNN,



(a) Previous target trees

(b) Predicted target trees

Figure 3.5: Previous and predicted target trees at time step $k = 70$

JIPDA-EHM1, and JIPDA-GM-EHM1, as can be seen in Figure 3.9. This happens because the OSPA metric was computed taking into account the radar cross-section, which introduces in the index the importance of keeping the correct disambiguating feature (and not swapping). On the other hand, the CPHD shows the ability to recover a low mean OSPA metric when close targets get separated. This is because when filtering the intensity function the CPHD filter discards nonsensical components, introduces new components (from the birth process), and carries over significant components to describe the whole target configuration in the scene. In just a few time steps after the target separation, the CPHD is able to use information from sensor S2 to correct its intensity function and outputs the intensity peaks that best explains the scene no matter the identities.

In terms of computational effort, we see from Figure 3.10 that the JPDA-EHM3 is more computationally complex than the other filters owing to the cost of maintaining (and restructuring) the dependencies between targets. The differences in runtime become more prominent when multiple targets take part on the dependency structure, but one should note that this complexity would not be prohibitive for most practical scenarios where pairwise dependencies are much more likely than multiple dependencies. Also, it is worth noting that the filters JIPDA-EHM1 and JIPDA-GM-EHM1 rely on a very computational efficient algorithm, EHM1, which potentiates computational efficiency of these filters. If compared to the classical JIPDA or JIPDA-GM, the JPDA-EHM3 would comparatively offset its own complexity by efficient computation of marginal association probabilities. The CPHD shows run times comparable to that of the JIPDA-GM-EHM1 since in this scenario the number of measurements and number

(a) Previous target trees

(b) Predicted target trees

Figure 3.6: Previous and predicted target trees at time step $k = 140$

of cardinality terms are moderately low, and most of the CPHD effort is applied to manage the intensity mixture components. As expected, the JIPDA-GNN is the fastest filter. The overall averaged run time per time step are 0.4436 s for the JPDA-EHM3, 0.1624 s for the CPHD, 0.0555 s for the JIPDA-GNN, 0.1211 s for the JIPDA-EHM1, and 0.1575 s for the JIPDA-GM-EHM1.

3.9 Conclusions

In this chapter, the challenge of representing target states and their mutual dependencies in a concise, accurate, and scalable way has been addressed. To that end, the hypothesis of independence between targets has been removed, and new hypotheses to structure the target dependencies have been adopted.

Under these assumptions, we have derived a new scalable multi-target tracking framework that considers dependency between target states and enables track management by inferring existence probabilities. The proposed algorithm models target dependency by probabilistic trees on which joint probability distributions of adjacent targets are calculated. Hence the method avoids maintaining a joint probability distribution over all the target states, which is infeasible except when the number of targets is small and enables a scalable algorithm for scenarios with a moderately high number of targets. For computing the marginal association probabilities, EHM 3 has been adopted allowing calculations for pairs of dependent targets.

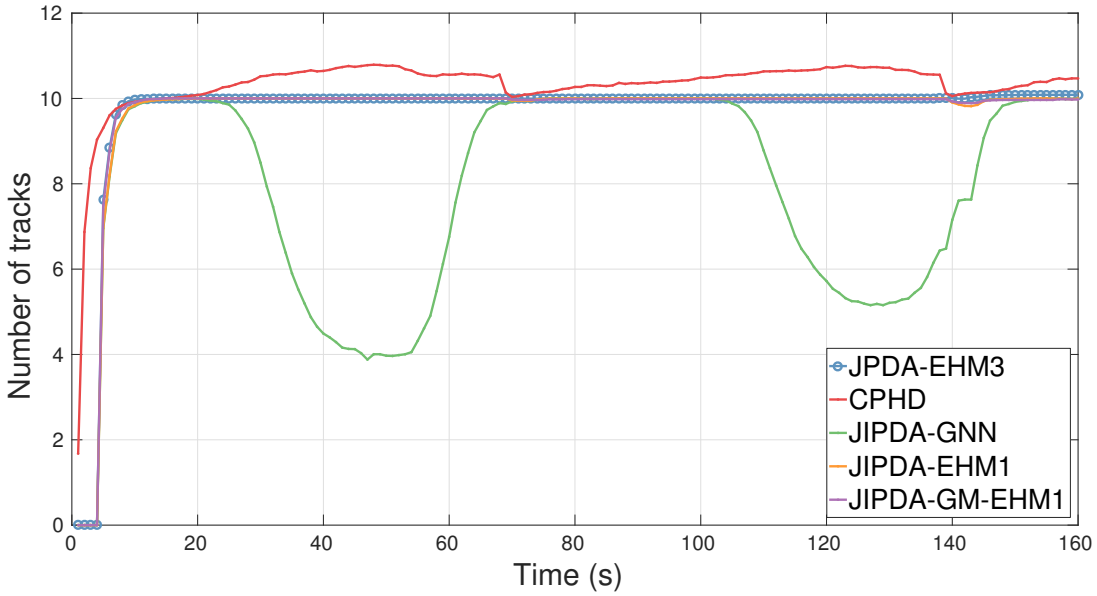


Figure 3.7: Average number of confirmed tracks

The numerical experiment presented made it clear that the proposed filter reduces the incidence of track swapping and substantially improves tracking capability when information sensible for association disambiguation are occasionally available. This is a direct consequence of the feature that allows the filter to maintain information of previous association ambiguities even when estimates have been accomplished, which means that representing target dependencies allows indirect inference where measurements for one target may provide information on others. This feature seems particularly useful in situations where one of two dependent targets becomes occluded, such that the information on the detected target allows to estimate the state of the occluded one.

The case made supports our argument that keeping target dependencies is clearly beneficial for scenarios of frequent association ambiguities, and that describing target dependencies by Bayesian probabilistic structures is accurate and more computationally feasible than maintaining the complete joint multi-target distribution.

References

[7] Y. Bar-Shalom, P. K. Willett, and X. Tian. *Tracking and Data Fusion: A Handbook of Algorithms*. YBS Publishing, Apr. 2011.

[15] H. A. P. Blom and E. A. Bloem. "Joint Particle Filtering of Multiple Maneuvering Targets From Unassociated Measurements". In: *Journal of Advances in Information Fusion* 1.1 (2006), pp. 15–34.

[16] H. A. P. Blom and E. A. Bloem. "Probabilistic data association avoiding track coalescence". In: *IEEE Transactions on Automatic Control* 45.2 (Feb. 2000), pp. 247–259.

[29] S. Challa et al. "Fundamentals of Object Tracking". In: Cambridge books online. Cambridge University Press, 2011. Chap. 5, pp. 161–167.

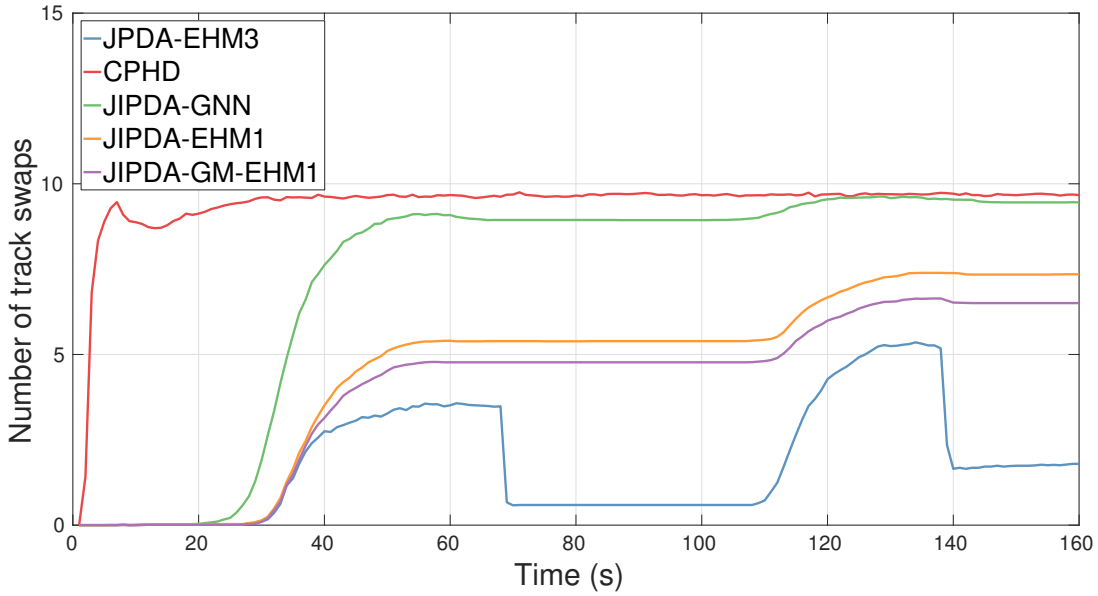


Figure 3.8: Average number of track swaps

[31] C. Chow and C. Liu. "Approximating Discrete Probability Distributions with Dependence Trees". In: *IEEE Transactions on Information Theory* 14.3 (May 1968), pp. 462–467.

[58] E. Delande, J. Houssineau, and D. E. Clark. "Multi-object filtering with stochastic populations". In: *ArXiv e-prints* (Jan. 2015). arXiv: 1501.04671 [stat.AP].

[73] T. E. Fortmann, Y. Bar-Shalom, and M. Scheffe. "Multi-target tracking using joint probabilistic data association". In: *19th IEEE Conference on Decision and Control including the Symposium on Adaptive Processes*. Dec. 1980, pp. 807–812.

[81] J. Goldberger, S. Gordon, and H. Greenspan. "An efficient image similarity measure based on approximations of KL-divergence between two Gaussian mixtures". In: *Proceedings of the Ninth IEEE International Conference on Computer Vision*. Vol. 1. Oct. 2003, pp. 487–493.

[94] P. R. Horridge and S. Maskell. "A scalable method of tracking targets with dependent distributions". In: *FUSION 2009, Proceedings of the 12th International Conference on Information Fusion*. July 2009, pp. 603–610.

[95] P. R. Horridge and S. Maskell. "Real-Time Tracking Of Hundreds Of Targets With Efficient Exact JPDAF Implementation". In: *FUSION 2006, Proceedings of the 9th International Conference on Information Fusion*. July 2006, pp. 1–8.

[117] R. A. Lau and J. L. Williams. "A structured mean field approach for existence-based multiple target tracking". In: *FUSION 2016, Proceedings of the 19th International Conference on Information Fusion*. July 2016, pp. 1111–1118.

[126] R. P. S. Mahler. "Multitarget Bayes Filtering via First-Order Multitarget Moments". In: *IEEE Transactions on Aerospace and Electronic Systems* 39.4 (Oct. 2003), pp. 1152–1178.

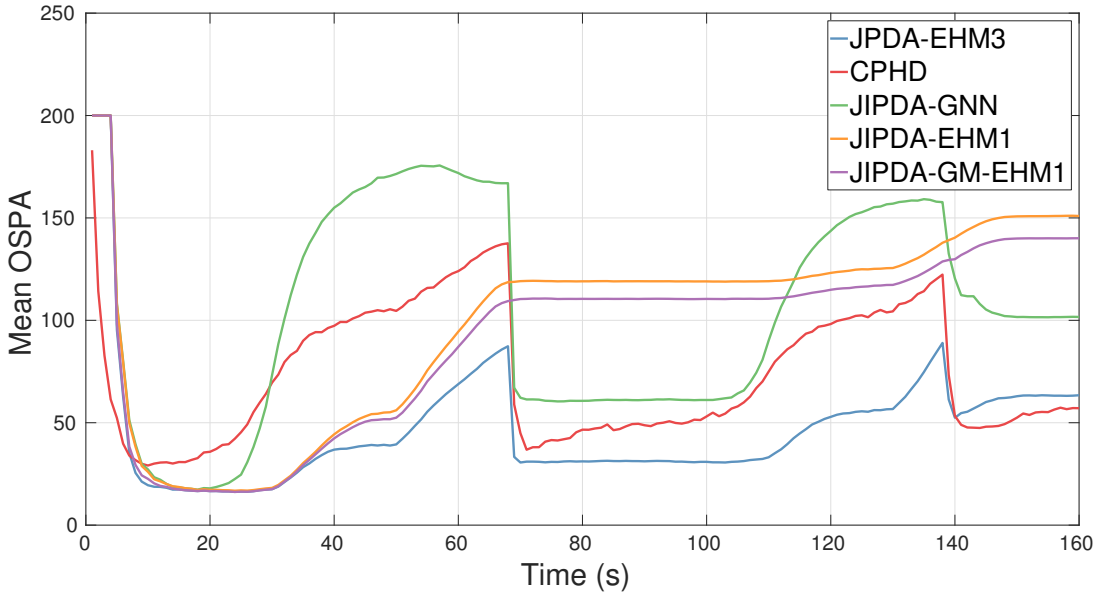


Figure 3.9: Mean OSPA metric over time

[129] R. P. S. Mahler. "PHD Filters of Higher Order in Target Number". In: *IEEE Transactions on Aerospace and Electronic Systems* 43.4 (Oct. 2007), pp. 1523–1543.

[131] R. P. S. Mahler. *Statistical Multisource-Multitarget Information Fusion*. Norwood, MA, USA: Artech House, Inc., 2007.

[135] S. Maskell, M. Briers, and R. Wright. "Fast Mutual Exclusion". In: *Proc. SPIE 5428, Signal and Data Processing of Small Targets 2004*. Vol. 5428. Aug. 2004, pp. 526–536.

[143] J. E. Moyal. "The General Theory of Stochastic Population Processes". In: *Acta Mathematica* 108 (1962), pp. 1–31.

[144] D. Musicki and R. Evans. "Joint Integrated Probabilistic Data Association - JIPDA". In: *FUSION 2002, Proceedings of the 5th International Conference on Information Fusion*. Vol. 2. July 2002, pp. 1120–1125.

[153] L. Y. Pao. "Multisensor multitarget mixture reduction algorithms for tracking". In: *Journal of Guidance, Control, and Dynamics* 17.6 (Nov. 1994), pp. 1205–1211.

[154] J. Pearl. "Reverend Bayes on inference engines: a distributed hierarchical approach". In: *AAAI-82, Proceedings of the 2nd National Conference on AI*. Pittsburg, PA, 1982, pp. 133–136.

[163] D. B. Reid. "An Algorithm for Tracking Multiple Targets". In: *IEEE Transactions on Automatic Control* 24.6 (Dec. 1979), pp. 843–854.

[164] S. Reuter et al. "The Labeled Multi-Bernoulli Filter". In: *IEEE Transactions on Signal Processing* 62.12 (June 2014), pp. 3246–3260.

[174] D. Schuhmacher, B.-T. Vo, and B.-N. Vo. "A Consistent Metric for Performance Evaluation of Multi-Object Filters". In: *IEEE Transactions on Signal Processing* 56.8 (Aug. 2008), pp. 3447–3457.

[188] R. Streit, C. Degen, and W. Koch. "The Pointillist Family of Multitarget Tracking Filters". In: *ArXiv e-prints* (May 2015). arXiv: 1505.08000 [stat.AP].

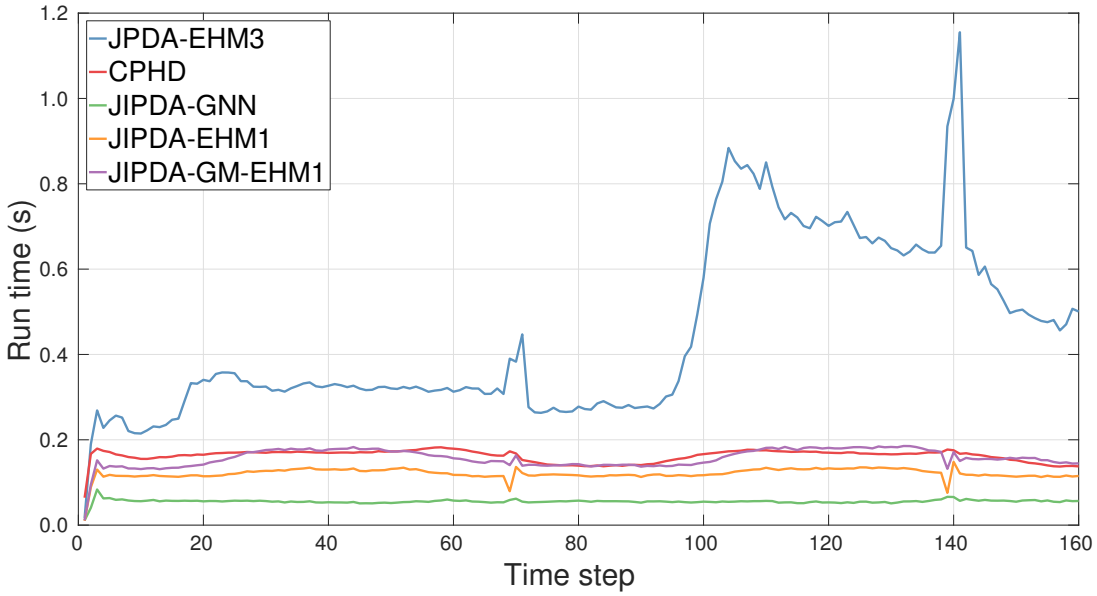


Figure 3.10: Average run time over time steps

[190] L. Svensson et al. "Set JPDA Filter for Multitarget Tracking". In: *IEEE Transactions on Signal Processing* 59.10 (Oct. 2011), pp. 4677–4691.

[202] B.-T. Vo, B.-N. Vo, and A. Cantoni. "The Cardinality Balanced Multi-Target Multi-Bernoulli Filter and Its Implementations". In: *IEEE Transactions on Signal Processing* 57.2 (Feb. 2009), pp. 409–423.

[203] B.-T. Vo, B.-N. Vo, and A. Cantoni. "Analytic Implementations of the Cardinalized Probability Hypothesis Density Filter". In: *IEEE Transactions on Signal Processing* 55.7 (July 2007), pp. 3553–3567.

[206] J. L. Williams. "Marginal multi-Bernoulli filters: RFS derivation of MHT, JIPDA, and association-based MeMBer". In: *IEEE Transactions on Aerospace and Electronic Systems* 51.3 (July 2015), pp. 1664–1687.

[207] J. L. Williams and R. A. Lau. "Data association by loopy belief propagation". In: *FUSION 2010, Proceedings of the 13th International Conference on Information Fusion*. July 2010, pp. 1–8.

[208] J. Williams and R. Lau. "Approximate evaluation of marginal association probabilities with belief propagation". In: *IEEE Transactions on Aerospace and Electronic Systems* 50.4 (Oct. 2014), pp. 2942–2959.

4

Discrete-Gamma Cardinalized Probability Hypothesis Density

In this chapter, our focus is on addressing the challenge of reducing computational complexity in multi-object estimation. In particular, this challenge is a fundamental problem in the multi-target tracking community, and it arises due to an increasing necessity of tracking enormous numbers of objects (hundredths to thousands) in applications that are very computationally demanding. To that end, in particular, we propose a filter based on a very effective simplification of a celebrated algorithm: the Cardinalized Probability Hypothesis Density (CPHD) filter.

The CPHD filter has become one of the most acclaimed algorithms for multi-target Bayesian filtering due to its ability to accurately estimate the number of objects and the object states in tracking scenarios affected by clutter. The CPHD filter generalizes the Probabilistic Hypothesis Density (PHD) filter by jointly propagating the first-order multi-target moment (intensity function) along with the entire probability distribution on the number of targets (cardinality distribution). In general, the CPHD recursion is computationally intractable, however, successful approximations have been devised with reported computational complexity dominated by $\mathcal{O}(m^3)$ operations per filtering iteration, where m is the number of measurements. Room for improvement was originally acknowledged by Mahler [130], who conceived the idea of approximating the cardinality distribution by two-parameter distributions. In this chapter, we further explore this idea to provide an efficient approximation of the CPHD where the cardinality distribution is modeled as a discretized Gamma distribution. Experiments show that the resulting filter is less computationally complex than the standard implementation of the CPHD filter but shows similar cardinality accuracy and variance.

4.1 Overview

Multi-target tracking is concerned with estimating the states of several objects of interest from a sequence of observations corrupted by noise, in the presence of missed detections and false alarms. The classical literature on multi-target tracking relies on the use of multivariate Bayesian statistics for estimating the target states while, at the same time, resorting to combinatorial analysis for capturing the very complex problem of associating objects to sequences of measurements. Celebrated methods, such as the Multi-Hypothesis Tracking (MHT) [163] and Joint Probabilistic Data Association (JPDA) [73] have become traditional due to three factors. First, their ability to incorporate established filtering techniques (e.g., Kalman measurement update). Second, their ability to provide accurate (and so useful) approximations of posterior multi-target densities. Third, the computational power available in modern digital computers

becoming sufficient to allow such methods to run in real-time.

The Probability Hypothesis Density (PHD) filter [126] has set the cornerstone for a completely new way of interpreting multi-target Bayesian inference: a view where data association is avoided by making a judicious choice related to the targets' state description while promoting a neat and elegant mathematical formulation. In this formulation, a collection of target states is considered as a random set-valued state and the collection of sensor measurements is treated as a random set-valued observation. While intimately related to the theory of stochastic population processes [143], the multi-target Bayesian framework proposed by Ronald Mahler has been rather stated in terms of random finite sets (RFS) on which operations are given in the context of finite-set statistics (FISST) [128, 131]. In this context, the PHD recursion [126] was developed as a first-moment approximation to the multi-target Bayesian filtering problem, where the algorithm propagates the posterior intensity function of the set of targets' states in time.

In their influential analysis of the PHD filter, Erdinc, Willett, and Bar-Shalom [68] pointed out that its most prominent limitation resides in that the filter is first-order on the number of targets (cardinality). Since the number of targets in the PHD filter is assumed to be Poisson-distributed, the cardinality mean equals the variance. This gives rise to very unstable cardinality estimates when considering scenarios involving a large number of missed detections or false alarms. In order to address this concern, Mahler derived the Cardinalized PHD (CPHD) filter [127, 129] which propagates the first-order multi-target moment (intensity function) along with the entire cardinality distribution. However, in general, practical implementations of the CPHD filter (e.g., [203]) are deemed to demand $\mathcal{O}(m^3)$ operations per filtering iteration, where m is the number of measurements. Mahler acknowledged there was room for computational improvement via additional approximation. With this potential in mind, he conceived the idea of approximating the cardinality distribution by two-parameter distributions and devised a cardinalized filter, the "binomial filter", whose number of targets is assumed to be distributed according to a binomial distribution [130]. Recent new developments in the same ethos have been proposed, such as that by Schlangen *et al.* [173] that assumes the cardinality distribution to be a Panjer distribution and establishes a recursion where both the first moment and a second-order entity, a regional variance, are propagated in the filtering procedure.

PHD filters that are second-order in target number have a clear advantage relative to the CPHD filter: they approximate a special case of the CPHD filter by reproducing, for a family of discrete probability distributions, the essential strategy of the Kalman filter, i.e., estimating sufficient statistics rather than the complete distribution. Mahler's binomial filter has particular merit in this context since the binomial distribution is the most basic discrete distribution that can describe underdispersed count phenomena, i.e., such that the variance can be smaller than the mean. Ther importance of this latter aspect becomes clear when we notice that, in most practical cases, cardinality estimates should be underdispersed because with a fairly high probability of detection and a moderate number of false alarms, the set of measurements is highly informative about the target number. Unfortunately, the binomial filter suffers from a fundamental drawback: the number of trials of a binomial distribution is strongly constrained by the filtering scenario, and the filter derived in [130] loses its validity when the predicted number of trials is smaller than the number of measurements.

This chapter proposes a new filter that is second-order in target number, where the targets'

state is assumed to follow an independent and identically distributed cluster process with the cardinality distribution modeled as a discretized Gamma distribution. Although apparently complicated, a discretized Gamma distribution allows simple calculations for approximating the first- and second-order moments of the posterior cardinality distribution, and efficiently addresses tracking scenarios with underdispersed target count while avoiding the restrictions imposed by the binomial filter. This latter feature is important because, when provided an i.i.d. cluster process with overdispersed cardinality model, the filter would most probably suffer from the same instability as that found in the PHD filter.

The chapter is organized as follows. Section 4.2 establishes some mathematical preliminaries and Section 4.3 briefly presents the CPHD recursion. Section 4.4 provides an analysis of the binomial filter. Section 4.5 proposes the Discrete-Gamma CPHD recursion. Section 4.6 presents a comparison of the proposed algorithm, the PHD filter, and the CPHD filter in the context of an illustrative simulated scenario. Section 4.7 concludes, and the Section 4.8 presents proofs associated with the derivation of the DG-CPHD algorithm.

4.2 Mathematical Preliminaries

In this section, we briefly present the mathematical concepts required to understand the chapter: multi-target statistics, point processes, probability generating functionals, and the intensity function of a state set density. We shall use the same description and notation originally proposed by Mahler [126, 129], though we note that an elegant description reminiscent of the seminal work by Moyal [143] has been increasingly adopted in recent literature [44, 45, 59, 57, 18, 173], which appear to the authors to have benefited from the measure-theoretic formalism. Our choice to adopt Mahler's description is just a matter of convenience when referring to expressions derived in [129], which may prevent the reader from having to translate between two equivalent formalisms.

4.2.1 Multi-target Statistics and Point Processes

Let $\mathrm{x}_i \in \mathcal{X} \subseteq \mathbb{R}^{d_x}$ be a d_x-dimensional vector describing the state of a single target identified by i. We assume a scenario with a random number $n \in \mathbb{N}$ of targets, and the collection of targets has no intrinsic ordering such that the joint state of this collection is represented by the finite set $\mathrm{x}_{1:n} = \{\mathrm{x}_1, \mathrm{x}_2, \ldots, \mathrm{x}_n\}$ where $\mathrm{x}_1, \ldots, \mathrm{x}_n$ are state vectors of all targets. By allowing such set of targets to be random in number of elements and state vectors, then the resulting set is a random finite set (RFS). Let Ξ be a random finite set and $\mathrm{X} \coloneqq \{\mathrm{x}_1, \ldots, \mathrm{x}_n\}$ its realization, the multi-target probability density can then be described as

$$p_{\Xi}(\mathrm{X}) = p_N(n)\, p_{\mathrm{x}_{1:n}|N}(\{\mathrm{x}_1, \ldots, \mathrm{x}_n\}|n) \tag{4.1}$$

where $p_N(n)$ is the cardinality distribution, and $p_{\mathrm{x}_{1:n}|N}(\{\mathrm{x}_1, \ldots, \mathrm{x}_n\}|n)$ is the joint probability density of the point set $\mathrm{x}_{1:n}$. In addition, the random finite set Ξ can be formally understood as a point process that pertains to the composite space $\mathscr{X} = \bigcup_{n=0}^{\infty} \mathcal{X}^n$, i.e., $\Xi \in \mathscr{X}$, with probability density p_{Ξ}. Particularly, if the point process Ξ is Poisson with mean μ, then $p_{\Xi}(\mathrm{X}) = e^{-\mu} \prod_{i=1}^{n} \mu p(\mathrm{x}_i)$, whereas for an i.i.d. cluster process the point-process density assumes the form $p_{\Xi}(\mathrm{X}) = p_N(n) \cdot n! \prod_{i=1}^{n} p(\mathrm{x}_i)$, where $p(\mathrm{x}_i)$ is the spatial probability density of each

target $i \in \{1, \ldots, n\}$ and the term $n!$ accounts for all possible orderings of the finite set of states $\mathrm{x}_{1:n}$. We can verify that $p_{\Xi}(\mathrm{X})$ is a probability density by identifying the value of its set integral over all possible realizations as

$$\begin{aligned}\int p_{\Xi}(\mathrm{X})\,\delta\mathrm{X} &\triangleq \sum_{n=0}^{\infty}\frac{1}{n!}\int_{\mathcal{X}^n} p_N(n)\,p_{\mathrm{x}_{1:n}|N}(\{\mathrm{x}_1,\ldots,\mathrm{x}_n\}|n)\;d(\mathrm{x}_1,\ldots,\mathrm{x}_n)\\ &= \sum_{n=0}^{\infty} p_N(n)\int_{\mathcal{X}^n}\left[\prod_{i=1}^{n} p(\mathrm{x}_i)\right]\;d(\mathrm{x}_1,\ldots,\mathrm{x}_n) = 1. \end{aligned} \tag{4.2}$$

In a similar way, let $\mathrm{y}_j \in \mathcal{Y} \subseteq \mathbb{R}^{d_y}$ be a d_y-dimensional vector describing the jth measurement collected by one (or more) sensor(s). We also assume that there is a random number $m \in \mathbb{N}$ of measurements, possibly originated from targets or false alarms and without specific order, that can be described by a random finite set Ψ with finite point set $\mathrm{y}_{1:m} = \{\mathrm{y}_1, \ldots, \mathrm{y}_m\}$ and where $\mathrm{y}_1, \ldots, \mathrm{y}_m$ are all observation vectors. For any realization of Ψ, say Y, multi-target likelihood functions can be defined as a generalization of single-target likelihoods with properties analogous to those described for the multi-target density, i.e., likelihoods of the type $p_{\Psi|\Xi}(\mathrm{Y}|\mathrm{X})$ are possible.

4.2.2 Probability Generating Functions and Functionals

Given a probability mass function $p_N(n) = \Pr\{N = n\}$ of a discrete non-negative random variable N, its probability generating function (p.g.f. or PGF) is defined as

$$G_N(\zeta) \triangleq \sum_{n=0}^{\infty} p_N(n)\zeta^n \equiv \mathbb{E}\left[\zeta^N\right], \tag{4.3}$$

which converges absolutely for $\zeta \in \mathbb{C}$ such that[1] $|\zeta| \leq 1$. From (4.3) one can recognize that $p_N(n) = G_N^{(n)}(0)/n!$ and $G_N(1) = 1$. Similarly, an analogous concept can be applied to a point-process random variable (random finite set), Ξ, via the probability generating functional (p.g.fl. or PGFL)

$$\begin{aligned} G_{\Xi}[h] &= \int h^{\mathrm{X}} p_{\Xi}(\mathrm{X})\,\delta\mathrm{X} \equiv \mathbb{E}\left[h^{\Xi}\right]\\ &= \sum_{n=0}^{\infty} p_N(n)\int_{\mathcal{X}^n}\left[\prod_{i=1}^{n} h(\mathrm{x}_i)p(\mathrm{x}_i)\right]\;d(\mathrm{x}_1,\ldots,\mathrm{x}_n), \end{aligned} \tag{4.4}$$

where $h : \mathcal{X} \to [0,1]$ is a test function analogous to ζ for the PGF, where $h^{\mathrm{X}} = 1$ if $\mathrm{X} = \emptyset$ and $h^{\mathrm{X}} = \prod_{i=1}^{n} h(\mathrm{x}_i)$ if $\mathrm{X} = \{\mathrm{x}_1, \ldots, \mathrm{x}_n\}$.

4.2.3 Functional Differentiation and the Intensity Function

The functional derivative of a probability generating functional with respect to $\mathrm{x} \in \mathcal{X}$ can be defined as

$$\frac{\delta G_{\Xi}}{\delta \mathrm{x}}[h] \triangleq \lim_{\varepsilon \searrow 0}\frac{G_{\Xi}[h + \varepsilon\delta_{\mathrm{x}}] - G_{\Xi}[h]}{\varepsilon}, \tag{4.5}$$

[1] Other values of ζ may lead to convergence, though $|\zeta| \leq 1$ is a sufficient condition.

where $\delta_{\mathrm{x}} = \delta(\mathrm{x}' - \mathrm{x})$ is the Dirac delta function concentrated at x and the differentiation is assumed linear and continuous for a fixed h. It is worth noting that definition (4.5) is heuristic because the Dirac delta is not a valid test function in view of 4.4. A rigorous definition of the functional derivative can be found in [125]. When differentiating with respect to a random set realization $\mathrm{X} \coloneqq \{\mathrm{x}_1, \ldots, \mathrm{x}_n\}$, one obtains

$$\frac{\delta G_{\Xi}}{\delta \mathrm{X}}[h] \triangleq \frac{\delta^n G_{\Xi}}{\delta \mathrm{x}_1 \ldots \delta \mathrm{x}_n}[h], \quad \frac{\delta G_{\Xi}}{\delta \emptyset}[h] \triangleq G_{\Xi}[h]. \tag{4.6}$$

As detailed in [131], functional derivatives obey rules analogous to those of elementary calculus, including the rule for a linear functional, the product rule, and the chain rule. The first moment of the random finite set Ξ (point process) is what is generally called the "intensity function", or the probability hypothesis density (PHD) [126]. The first moment can be derived from the first derivative of the probability generating functional as

$$D_{\Xi}(\mathrm{x}) = \frac{\delta G_{\Xi}}{\delta \mathrm{x}}[1], \tag{4.7}$$

or equivalently from a set expectation according to

$$D_{\Xi}(\mathrm{x}) = \int \delta_{\mathrm{X}}(\mathrm{x}) p_{\Xi}(\mathrm{X}) \delta \mathrm{X} = \mathbb{E}\left[\delta_{\Xi}(\mathrm{x})\right], \tag{4.8}$$

where $\delta_{\mathrm{X}}(\mathrm{x}) \triangleq \sum_{\hat{\mathrm{x}} \in \mathrm{X}} \delta(\mathrm{x} - \hat{\mathrm{x}})$. The intensity function $D_{\Xi}(\mathrm{x})$ can be interpreted as a density of objects at x, i.e., $\mathbb{E}\left[|\Xi \cap \mathcal{S}|\right] = \int_{\mathcal{S}} D_{\Xi}(\mathrm{x}) d\mathrm{x}$ gives the expected number of objects in the region $\mathcal{S} \subseteq \mathcal{X}$.

4.3 The Cardinalized PHD Filter

In this section, we present the CPHD filter in its original form. The CPHD recursion was proposed by Mahler in [127, 129] to address the instability of the PHD filter when estimating the target number. The CPHD recursion propagates the first-order multi-target moment (intensity function) along with the entire cardinality distribution. The CPHD recursion is derived based on the following assumptions:

Assumption 4.1. *Each target moves independently of one another, with motion modeled by a single-target Markov transition density $p_{t,k|k-1}(\mathrm{x}|\mathrm{x}')$, which we abbreviate as $p_t(\mathrm{x}|\mathrm{x}')$.*

Assumption 4.2. *Existing targets may disappear between two time steps with probability $1 - p_{s,k|k-1}(\mathrm{x})$, where $p_{s,k|k-1}(\mathrm{x})$ is the probability that a target with state x will survive between time steps $k-1$ and k, hereafter abbreviated as $p_s(\mathrm{x})$.*

Assumption 4.3. *New targets can appear in the scene independently of the existing targets, according to a Poisson point process. New targets are realized at time step k with joint density $b_{k|k-1}(\mathrm{X})$. The intensity function of the corresponding random finite set is denoted as $D_b(\mathrm{x})$, and its cardinality distribution is denoted as $p_b(n)$ whose p.g.f. is $G_b(x)$.*

Assumption 4.4. *Measurements generated from targets are independent of one another, with single-target likelihood function $\ell_{k,\mathrm{y}}(\mathrm{x}) = p_{\ell,k}(\mathrm{y}|\mathrm{x})$, hereafter abbreviated as $\ell_{\mathrm{y}}(\mathrm{x})$.*

Assumption 4.5. *The sensor detects a single target with state* x *at the time step* k *with probability* $p_{d,k}(\mathrm{x})$*, abbreviated as* $p_d(\mathrm{x})$ *and the probability of not detecting the target is denoted as* $q_d(\mathrm{x}) = 1 - p_d(\mathrm{x})$.

Assumption 4.6. *False alarms may affect the observation, and are assumed to be characterized by a Poisson random finite set (or point process) and independent of the actual measurements. At the time step* k*, the sensor obtains a number of false alarms whose spatial distribution is individually given by the probability density* $c_k(\mathrm{y})$ *and whose cardinality distribution is given by* $p_{c,k}(m)$*, abbreviated as* $c(\mathrm{y})$ *and* $p_c(m)$ *respectively, and where the average number of false alarms is denoted by* $\lambda = \mathbb{E}_{p_c}[m]$*. The p.g.f. of* $p_c(m)$ *is denoted as* $G_c(x)$.

Assumption 4.7. *Both the prior and posterior multi-target random finite sets are i.i.d. cluster processes.*

4.3.1 CPHD Prediction Step

At a given time instant $k-1$, one has in possession estimates of the intensity $D_{k-1}(\mathrm{x}|\mathrm{Y}_{1:k-1})$, the expected number of targets $\hat{N}_{k-1}$, and the cardinality distribution $p_{k-1}(n|\mathrm{Y}_{1:k-1})$, conditioned on all measurements received to date, $\mathrm{Y}_{1:k-1} = \{\mathrm{Y}_1, \ldots, \mathrm{Y}_{k-1}\}$, where Y_{k-1} is the RFS realization containing all observations received at time instant $k-1$. In this section, we suppress the conditioning on $\mathrm{Y}_{1:k-1}$ to express the intensity function and cardinality distribution in the concise forms $D_{k-1}(\mathrm{x})$ and $p_{k-1}(n)$, respectively.

We write $\varsigma_{k-1}(\mathrm{x}) \coloneqq \hat{N}_{k-1}^{-1} D_{k-1}(\mathrm{x})$, abbreviated as $\varsigma(\mathrm{x})$ in this section, and recall the definition of inner product between functions as $\langle f, g\rangle \triangleq \int_{\mathcal{X}} f(\mathrm{x}')g(\mathrm{x}')d\mathrm{x}'$. The prior p.g.f. corresponding to the prior cardinality distribution is given by

$$G_{k|k-1}(x) = G_b(x) \cdot G_{k-1}(\langle 1 - p_s, \varsigma\rangle + \langle p_s, \varsigma\rangle x), \tag{4.9}$$

where $G_{k-1}(x)$ is the p.g.f. of the cardinality distribution at time step $k-1$, and the equality[2] is valid under *Assumption 4.7*. Expression (4.9) follows from the assumption that the birth process is independent of the prior process of targets that survived, which in turn is written for a Bernoulli survival transition by using the Watson-Galton recursion [205]. The CPHD prediction step obtains the prior intensity function, prior expected number of targets, and prior cardinality distribution according to

$$D_{k|k-1}(\mathrm{x}) = D_b(\mathrm{x}) + \int_{\mathcal{X}} p_s(\mathrm{x}') \cdot p_t(\mathrm{x}|\mathrm{x}')D_{k-1}(\mathrm{x}')d\mathrm{x}', \tag{4.10}$$

$$\hat{N}_{k|k-1} = \hat{N}_{b,k} + \hat{N}_{s,k}, \tag{4.11}$$

$$p_{k|k-1}(n) = \sum_{i=0}^{n} p_b(n-i)\frac{1}{i!}G_{k-1}^{(i)}(\langle 1 - p_s, \varsigma\rangle)\langle p_s, \varsigma\rangle^i, \tag{4.12}$$

where $\hat{N}_{b,k} = \langle 1, D_b\rangle$ is the expected number of newborn targets, and $\hat{N}_{s,k} = \langle p_s, D_{k-1}\rangle$ is the expected number of targets that have survived from time step $k-1$.

[2] In practice this is an approximation.

4.3.2 CPHD Measurement Update

In this section, we assume that measurements are collected from a single sensor at time instant k, as a realization Y_k of the observation RFS Ψ_k, with a finite point set $\mathrm{y}_{k,1:m_k} = \{\mathrm{y}_{k,1}, \ldots, \mathrm{y}_{k,m_k}\}$ of collected measurements. Based on the prior intensity function $D_{k|k-1}(\mathrm{x}|\mathrm{Y}_{1:k-1})$, the prior expected number of targets $\hat{N}_{k|k-1}$, and the prior cardinality distribution $p_{k|k-1}(n|\mathrm{Y}_{1:k-1})$, the realization Y_k is used to produce the posterior intensity function $D_k(\mathrm{x}|\mathrm{Y}_{1:k})$, the posterior expected number of targets $\hat{N}_k$, and the posterior cardinality distribution $p_k(n|\mathrm{Y}_{1:k})$.

We denote the intensity functions as $D_{k|k-1}(\mathrm{x})$ and $D_k(\mathrm{x})$, and the cardinality distributions as $p_{k|k-1}(n)$ and $p_k(n)$. In addition, we set $\varsigma_{k|k-1}(\mathrm{x}) \coloneqq \hat{N}_{k|k-1}^{-1} D_{k|k-1}(\mathrm{x})$, abbreviated as $\varsigma(\mathrm{x})$ in this section. The p.g.f. of the posterior cardinality distribution is approximately given by

$$G_{k|k}(x) = \frac{\sum_{j=0}^{m_k} x^j G_c^{(m_k-j)}(0) G^{(j)}(x\langle q_d, \varsigma\rangle)\sigma_j(\mathrm{Y}_k)}{\sum_{i=0}^{m_k} G_c^{(m_k-i)}(0) G^{(i)}(\ \langle q_d, \varsigma\rangle)\sigma_i(\mathrm{Y}_k)}, \tag{4.13}$$

where $m_k = |\mathrm{Y}_k|$, $G(x) = G_{k|k-1}(x)$, and for a set Y with $m = |\mathrm{Y}|$,

$$\sigma_i(\mathrm{Y}) \triangleq \sigma_{m,i}\left(\frac{\langle p_d \ell_{\mathrm{y}_1}, \varsigma\rangle}{c(\mathrm{y}_1)}, \ldots, \frac{\langle p_d \ell_{\mathrm{y}_m}, \varsigma\rangle}{c(\mathrm{y}_m)}\right), \tag{4.14}$$

where $\sigma_{m,i}(x_1, \ldots, x_m)$ is the elementary homogeneous symmetric function of degree i in $x_1, \ldots, x_m$. By defining

$$\Upsilon_k[\mathrm{Y}] = \frac{\sum_{j=0}^{|\mathrm{Y}|} G_c^{(|\mathrm{Y}|-j)}(0) G^{(j+1)}(\langle q_d, \varsigma\rangle)\sigma_j(\mathrm{Y}\ )}{\sum_{i=0}^{m_k} G_c^{(m_k-i)}(0) G^{(i)}\quad (\langle q_d, \varsigma\rangle)\sigma_i(\mathrm{Y}_k)}, \tag{4.15}$$

the CPHD measurement-update step can be described as

$$D_k(\mathrm{x}) = \frac{q_d(\mathrm{x})}{G^{(1)}(1)}\Upsilon_k\left[\mathrm{Y}_k\right] D_{k|k-1}(\mathrm{x}) + \frac{p_d(\mathrm{x})}{G^{(1)}(1)} \sum_{\mathrm{y}\in\mathrm{Y}_k} \frac{\ell_{\mathrm{y}}(\mathrm{x})}{c(\mathrm{y})}\Upsilon_k\left[\mathrm{Y}_k \setminus \{\mathrm{y}\}\right] D_{k|k-1}(\mathrm{x}), \tag{4.16}$$

$$\hat{N}_k = G_k^{(1)}(1) \approx \arg\max_n p_k(n), \tag{4.17}$$

$$p_k(n) = \frac{\sum_{j=0}^{m_k} G_c^{(m_k-j)}(0) \frac{\langle q_d, \varsigma\rangle^{n-j}}{(n-j)!} \cdot G^{(j)(n-j)}(0)\sigma_j(\mathrm{Y}_k)}{\sum_{i=0}^{m_k} G_c^{(m_k-i)}(0) G^{(i)}(\langle q_d, \varsigma\rangle)\sigma_i(\mathrm{Y}_k)}, \tag{4.18}$$

where $G^{(j)(e)} = D_x^e G^{(j)}$, where D_x^e is Euler's notation for differentiation.

4.4 The Binomial Filter

The binomial filter [130] was proposed as one further step towards simplification of the CPHD filter. The strategy employed was to mimic, for a hypothesized discrete distribution, the procedure of a Kalman filter for the cardinality random variable, i.e., estimating sufficient statistics. Mahler proposed that the cardinality should be distributed according to a binomial distribution,

$$p_N(n) = \binom{\nu}{n}\omega^n(1-\omega)^{\nu-n},$$

which models the probability of n successes out of ν trials each with probability ω. For the purpose of counting objects in a scene, the binomial distribution has two virtues: (i) underdispersion, i.e., the variance is smaller than or equal to the mean, and (ii) simplicity, with an analytic probability generating function $G_{bin}(\zeta) = (1 - \omega + \omega\zeta)^{\nu}$. As touched on before, the first virtue is very important for the vast majority of practical applications since in scenarios of a fairly high probability of detection and a moderate number of false alarms, a set of measurements is highly informative about target number. Under these conditions, for a sufficiently high number of targets, it is very unlikely that the cardinality variance will be perceived as greater than the expected target number. To verify the underdispersed characteristic of the binomial distribution, one just needs to note that, for $0 \leq \omega \leq 1$, $\text{var}(N) = \nu\omega(1-\omega) \leq \nu\omega = \mathbb{E}[N]$ because $1 - \omega \leq 1$.

On the other hand, the binomial filter has a fundamental problem: for a given number of measurements m_k received at time step k, the filter requires that $\nu_{k|k-1} \geq m_k$. This is actually an essential assumption associated with *Theorem 2* in [130], which makes it possible to state the measurement-update step. Ultimately, this assumption poses a limitation in the number of newly appearing targets that the binomial filter can cope with. Supposing that $\hat{N}_{k|k-1}$ targets have been predicted in the scene and no false alarms or missed detections took place, then the maximum number of new targets that the binomial filter could account for would be[3] $\hat{\nu}_{k|k-1} - \hat{N}_{k|k-1} \approx \text{var}(N_{k|k-1})/\hat{\omega}_{k|k-1}$. The problem arises because when predicting $\nu_{k|k-1}$ and $\omega_{k|k-1}$ no information on the actual number of new targets is available, and the problem tends to be exacerbated when high number of false alarms may be realized, i.e., most probably $\hat{\nu}_{k|k-1} \geq m_k$ would not hold.

4.5 The Discrete-Gamma CPHD Filter

In this section, we propose a new approximation to the CPHD in the same ethos of the binomial filter. Such approximation is based on a new cardinality model, obtained by discretizing the Gamma distribution for describing count data. This cardinality model and its properties are explained in Sections 4.5.1 and 4.5.2. In Section 4.5.3 we present the prediction step of the proposed filter, and in Section 4.5.4 we present its measurement-update step. Section 4.5.5 is dedicated to an implementation of the Discrete-Gamma CPHD filter based on Gaussian mixtures.

4.5.1 The Discrete Gamma Distribution

In order to derive the new filter, we assume that the prior and posterior cardinality distributions can be accurately approximated as a discretized form of the Gamma distribution, called *discrete*

[3] Typically $\hat{\omega}_{k|k-1}$ is of the same order of magnitude as $\text{var}(N_{k|k-1})$.

Gamma (*dGamma*) hereafter, with probability mass function given by

$$\begin{aligned} dGamma(n|\alpha,\beta) &= \frac{Gamma(n|\alpha,\beta)}{\sum_{u=0}^{\infty} Gamma(u|\alpha,\beta)}, \quad n \in \mathbb{N}_0, \\ dGamma(n|\alpha,\beta) &= \frac{\Gamma(\alpha)^{-1}\beta^{\alpha} n^{\alpha-1} e^{-\beta n}}{\Gamma(\alpha)^{-1}\beta^{\alpha} \sum_{u=0}^{\infty} u^{\alpha-1} e^{-\beta u}} \\ &= \frac{n^{\alpha-1} e^{-\beta n}}{\sum_{u=0}^{\infty} u^{\alpha-1} e^{-\beta u}}. \end{aligned} \tag{4.19}$$

where $Gamma(x|\alpha,\beta)$ denotes the continuous Gamma probability density at $x \in \mathbb{R}_+$ with a shape parameter, $\alpha > 0$, and a rate parameter, $\beta > 0$. Note that (4.19) is built by multiplying the continuous Gamma probability density by a Dirac comb with unit period, $\text{Ш}(x) = \sum_{n\in\mathbb{Z}} \delta(x-n)$, while accounting for a normalization factor, i.e., $dGamma(n|\alpha,\beta) = \mathfrak{N}^{-1} Gamma(x|\alpha,\beta)\text{Ш}(x)$, where the normalizing constant is calculated by[4]

$$\begin{aligned} \mathfrak{N} &= \int_{\mathbb{R}} Gamma(x|\alpha,\beta)\text{Ш}(x)dx \\ &= \sum_{u\in\mathbb{N}_0} \int_{\mathbb{R}_+} Gamma(x|\alpha,\beta)\delta(x-u)dx \\ &= \sum_{u\in\mathbb{N}_0} Gamma(u|\alpha,\beta). \end{aligned} \tag{4.20}$$

The utility of such probability mass function will become evident later on when we present the filter equations that follow from it. For now it suffices to mention its benign characteristics, involving a simple cardinality model that does not suffer from the shortcoming shown for the binomial filter, and being suitable for inference of count phenomena. Specifically, the discrete Gamma distribution can be either underdispersed and overdispersed. The discrete Gamma distribution has the probability generating function

$$\begin{aligned} G_{dGamma}(\zeta) &= \sum_{n=0}^{\infty} dGamma(n|\alpha,\beta) \cdot \zeta^n \\ &= \frac{\sum_{n=0}^{\infty} n^{\alpha-1}(e^{-\beta}\zeta)^n}{\sum_{u=0}^{\infty} u^{\alpha-1} e^{-\beta u}} \\ &= \frac{\text{Li}_{1-\alpha}(e^{-\beta}\zeta)}{\text{Li}_{1-\alpha}(e^{-\beta}\ )}, \end{aligned} \tag{4.21}$$

where $\text{Li}_s(z) \triangleq \sum_{k=1}^{\infty} k^{-s} z^k$, for $|z| < 1$, is the function known as the *polylogarithm* of order $s \in \mathbb{C}$ and argument $z \in \mathbb{C}$. The polylogarithm function has great importance in number theory, representing a number of transcendental functions (e.g., Riemann zeta function). Interesting particular cases involve $\text{Li}_1(z) = -\log(1-z)$, and

$$\text{Li}_{-\ell}(z) = \frac{1}{(1-z)^{\ell+1}} \sum_{i=0}^{\ell} A(\ell,i) z^{\ell-i}, \quad \ell \in \mathbb{Z}_+, \tag{4.22}$$

[4]Recall that the support of the Gamma probability density is $\mathbb{R}_+$.

where $A(\ell, i)$ is an Eulerian number[5], and note that (4.22) can be computed by evaluating a finite number of terms. However, in practice, as ℓ increases, computing $A(\ell, i)$ becomes computationally expensive and increasingly prone to round-off errors. Resorting to the known identity $\mathrm{Li}_{-\ell}(z) = (z \cdot \partial_z)^{\ell} [z/(1-z)]$ could also be an alternative, but the numerical evaluation of these rational expressions increasingly suffers from cancellation errors as ℓ becomes large. In those cases, truncating the polylogarithm's definition directly may give a better answer [209].

Before closing this section it is worth remarking that other discrete distributions for count data were investigated. At a first glance, some of them appeared as suitable candidates for our endeavor, including the well known generalized Poisson distribution [32] and, more generally, the class of Lagrangian distributions [33], the Conway-Maxwell-Poisson distribution [176], and other forms of Gamma-count distributions [27]. In our investigation, we have verified the feasibility of such distributions, observing different disadvantages in each one them: the Lagrangian distributions propose very complex forms of probability generating functions (based on the series expansions of a *Lagrangian transformation*), the Conway-Maxwell-Poisson distribution requires approximations for the moments that are only valid for specific conditions on the parameters, and the Gamma-count distribution in [27] results in a probability generating function involving infinitely many evaluations of the incomplete Gamma function. Eventually, we found more favorable characteristics in our approach.

4.5.2 Moments of the Discrete Gamma Distribution

From (4.21), it is not clear whether closed-form solutions exist for the discrete Gamma factorial moments. We will rely on a procedure closely related to that shown in [3] in the context of inducing the Euler-Mclaurin summation formula. First observe that

$$\sum_{n=0}^{\infty} n^{\alpha-1} e^{-\beta n} \triangleq \lim_{m\to\infty} \sum_{n=0}^{m} n^{\alpha-1} e^{-\beta n}$$

converges for $\beta > 1$ according to the ratio criterion. Then, we shall assume $\beta > 1$ from now on. The analysis requires the distribution maximum, $x_m = \arg\max_x \mathit{Gamma}(x|\alpha, \beta)$, $x_m \neq 0$, given by

$$\frac{d}{dx}\left(\frac{\beta^{\alpha} x^{\alpha-1} e^{-\beta x}}{\Gamma(\alpha)}\right)_{x_m} = \frac{\beta^{\alpha} x_m^{\alpha-1} e^{-\beta x}}{\Gamma(\alpha)} \left[(\alpha-1)x_m^{-1} - \beta\right] = 0,$$
$$x_m = (\alpha-1)\beta^{-1}. \tag{4.23}$$

The first two moments of $\mathit{dGamma}(n|\alpha, \beta)$ are evaluated by

$$\mu_N = G^{(1)}_{dGamma}(1) = \frac{\mathrm{Li}_{-\alpha}(e^{-\beta})}{\mathrm{Li}_{1-\alpha}(e^{-\beta})}, \tag{4.24}$$
$$\sigma_N^2 = G^{(2)}_{dGamma}(1) - \mu_N^2 + \mu_N. \tag{4.25}$$

In order to evaluate these moments, one must approximate ratios of polylogarithms when computing $G^{(1)}_{dGamma}(1)$ and $G^{(2)}_{dGamma}(1)$, whose exact values are not easily calculable. We shall not prove that $G^{(1)}_{dGamma}(1) \approx \alpha\beta^{-1}$ and $G^{(2)}_{dGamma}(1) \approx \alpha(\alpha+1)\beta^{-2} - \alpha\beta^{-1}$ to first order

[5]The Eulerian number $A(\ell, i)$ is the number of permutations of the numbers 1 to ℓ in which i elements are greater than the previous element.

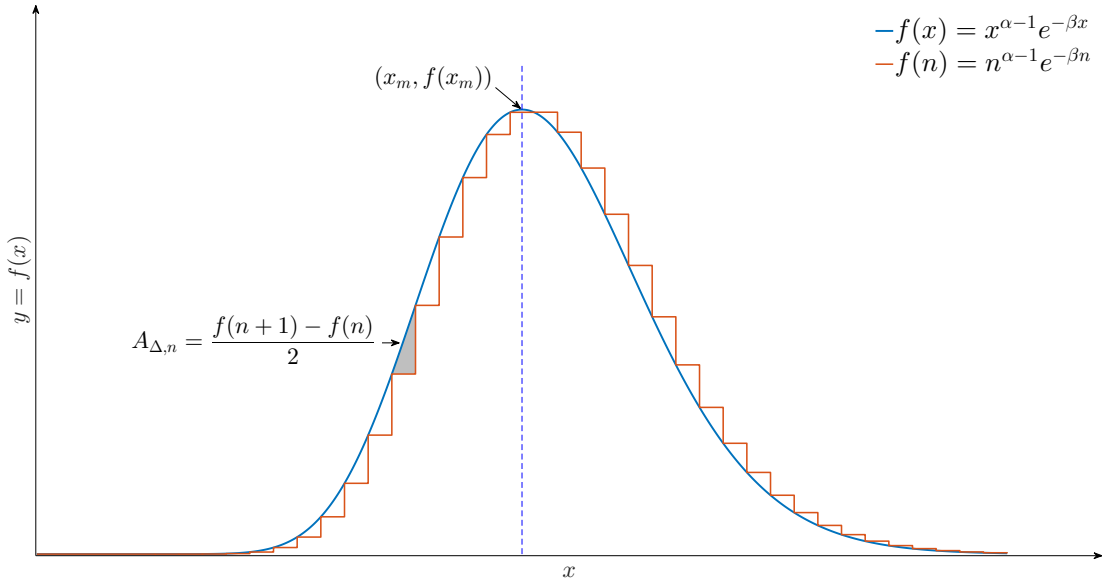


Figure 4.1: Differences between summation and integral of $x^{\alpha-1}e^{-\beta x}$

precision, but rather provide an intuitive argument that hopefully will convince the reader that higher order terms may be neglected. Set $f(x) \coloneqq x^{\alpha-1}e^{-\beta x}$ and refer to Figure 4.1 to observe that

$$d_{n<x_m} \coloneqq \sum_{n=0}^{\lfloor x_m \rfloor - 1} f(n) - \int_0^{\lfloor x_m \rfloor} f(x)dx < 0,$$
$$d_{n>x_m} \coloneqq \sum_{n=\lfloor x_m \rfloor}^{\infty} f(n) - \int_{\lfloor x_m \rfloor}^{\infty} f(x)dx > 0,$$

where $\lfloor x \rfloor$ is the greatest integer less than or equal to x. Let us compute $d_{n<x_m}$ and $d_{n>x_m}$ by a first order approximation where the difference $f(n) - f(x)$, integrated within each interval $x \in \mathcal{I}_n = (n, n+1)$, is approximated by the area of a triangle (Figure 4.1), $A_{\Delta,n}$, with base $\Delta x_n = 1$ and height $\Delta f_n = |f(n+1) - f(n)|$, to result

$$d_{n<x_m} \approx -\frac{1}{2} \sum_{n=1}^{\lfloor x_m \rceil} (f(n) - f(n-1)) = -\frac{f(\lfloor x_m \rceil)}{2},$$
$$d_{n>x_m} \approx +\frac{1}{2} \sum_{n=\lfloor x_m \rceil}^{\infty} (f(n) - f(n+1)) = +\frac{f(\lfloor x_m \rceil)}{2},$$

since $f(0) = \lim_{x\to\infty} f(x) = 0$, and where $\lfloor x \rceil$ is the closest integer to x. Therefore, we can approximate the total difference to first order as

$$d_n = d_{n<x_m} + d_{n>x_m} \approx \frac{f(\lfloor x_m \rceil) - f(\lfloor x_m \rceil)}{2} = 0,$$

which allows us to get

$$\begin{aligned}\mathrm{Li}_{1-\alpha}(e^{-\beta}) = \sum_{n=0}^{\infty} n^{\alpha-1}e^{-\beta n} &= \int_0^{\infty} x^{\alpha-1}e^{-\beta x}dx + d_n \\ &\approx \beta^{-\alpha}\Gamma(\alpha). \end{aligned} \tag{4.26}$$

If one is interested in a higher-order approximation of d_n, the areas shown in Figure 4.1 should rather be approximated by integrals, within each interval $\mathcal{I}_n$, of a higher-order fitting polynomial. As a direct consequence of (4.26) and based on the property $\Gamma(\alpha+\ell) = (\alpha+\ell-1)\Gamma(\alpha+\ell-1)$, $\ell \in \mathbb{N}$, one can easily verify the validity of the approximations:

$$\mu_N = \frac{\mathrm{Li}_{-\alpha}(e^{-\beta})}{\mathrm{Li}_{1-\alpha}(e^{-\beta})} \approx \frac{\beta^{-(\alpha+1)}\Gamma(\alpha+1)}{\beta^{-\alpha}\Gamma(\alpha)} = \alpha\beta^{-1}, \tag{4.27}$$

$$\begin{aligned}\sigma_N^2 &= \frac{\mathrm{Li}_{-\alpha-1}(e^{-\beta}) - \mathrm{Li}_{-\alpha}(e^{-\beta})}{\mathrm{Li}_{1-\alpha}(e^{-\beta})} - \mu_N^2 + \mu_N \\ &\approx \frac{\beta^{-(\alpha+2)}\Gamma(\alpha+2) - \beta^{-(\alpha+1)}\Gamma(\alpha+1)}{\beta^{-\alpha}\Gamma(\alpha)} - \mu_N^2 + \mu_N \\ &= \frac{\beta^{-\alpha}\Gamma(\alpha)\left[\alpha(\alpha+1)\beta^{-2} - \alpha\beta^{-1}\right]}{\beta^{-\alpha}\Gamma(\alpha)} - \mu_N^2 + \mu_N \\ &= \alpha\beta^{-2}. \end{aligned} \tag{4.28}$$

4.5.3 Discrete-Gamma CPHD Prediction

The same assumptions used to formulate the CPHD filter, established Section 4.3, shall be applied to derive the Discrete-Gamma CPHD recursion. In addition to *Assumptions 4.1 to 4.7*, the following premise is also considered:

Assumption 4.8. *The prior and posterior multi-target random finite sets are assumed with multi-object discrete-Gamma probability densities of the form*

$$p_{\Xi}(\mathrm{X}) = dGamma(n) \cdot n! \prod_{i=1}^{n} p(\mathrm{x}_i), \tag{4.29}$$

where $p(\mathrm{x}_i)$ is the spatial probability density of each object.

The probability generating functional of (4.29) can be obtained as

$$\begin{aligned} G_{\Xi}[h] &= \mathbb{E}_{\Xi}\left[e^{\sum_{i=0}^{n}\log h(\mathrm{x}_i)}\right] \\ &= \sum_{n=0}^{\infty} p_N(n) \int_{\mathcal{X}^n} e^{\sum_{i=0}^{n}\log h(\mathrm{x}_i)} \left[\prod_{i=1}^{n} p(\mathrm{x}_i)\right] d\mathrm{x}_{1:n} \\ &= \sum_{n=0}^{\infty} dGamma(n) \int_{\mathcal{X}^n} \left[\prod_{i=1}^{n} h(\mathrm{x}_i)p(\mathrm{x}_i)\right] d\mathrm{x}_{1:n} \\ &= \sum_{n=0}^{\infty} \frac{n^{\alpha-1}\left[e^{-\beta}\int_{\mathcal{X}} h(\mathrm{x})\varsigma(\mathrm{x})\,d\mathrm{x}\right]^n}{\mathrm{Li}_{-\alpha+1}(e^{-\beta})} \\ &= \frac{\mathrm{Li}_{-\alpha+1}(e^{-\beta}\langle h,\varsigma\rangle)}{\mathrm{Li}_{-\alpha+1}(e^{-\beta})}, \end{aligned} \tag{4.30}$$

for $\beta \in \mathbb{R}$, $\beta > 1$, and where $\varsigma(\mathrm{x}_i) \coloneqq p(\mathrm{x}_i)$.

Now, based on previous knowledge of D_{k-1}, α_{k-1} and β_{k-1}, i.e., the posterior intensity function and cardinality parameters from the previous filtering iteration, the prediction step shall obtain the predicted intensity, $D_{k|k-1}$, and predicted cardinality parameters, $\alpha_{k|k-1}$ and $\beta_{k|k-1}$, which are functions of the prior cardinality mean, $\mu_{N,k|k-1}$, and of the prior cardinality variance, $\sigma^2_{N,k|k-1}$. The following proposition establishes how the predicted intensity and cardinality parameters are achieved. Proofs are presented in the Section 4.8.

Proposition 4.1. *Provided the posterior intensity function at time instant $k-1$, D_{k-1}, and the parameters of a posterior discrete-Gamma cardinality distribution at time instant $k-1$, α_{k-1} (shape) and β_{k-1} (rate), then, under Assumptions 4.1–4.3 and 4.8, the predicted intensity function, $D_{k|k-1}$, and the predicted cardinality parameters, $\alpha_{k|k-1}$ and $\beta_{k|k-1}$, are given by*

$$D_{k|k-1}(\mathrm{x}) = D_b(\mathrm{x}) + \int_{\mathcal{X}} p_s(\mathrm{x}') \cdot p_t(\mathrm{x}|\mathrm{x}') D_{k-1}(\mathrm{x}') d\mathrm{x}', \tag{4.31}$$

$$\alpha_{k|k-1} \approx \frac{\mu^2_{N,k|k-1}}{\sigma^2_{N,k|k-1}},\ \beta_{k|k-1} \approx \frac{\mu_{N,k|k-1}}{\sigma^2_{N,k|k-1}}, \tag{4.32}$$

where

$$\mu_{N,k|k-1} = \hat{N}_{k|k-1} = \hat{N}_{b,k} + \overbrace{\alpha_{k-1}\beta^{-1}_{k-1}\langle p_s, \varsigma\rangle}^{\hat{N}_{s,k}}, \tag{4.33}$$

$$\sigma^2_{N,k|k-1} \approx \mu_{N,k|k-1} + \langle p_s, \varsigma\rangle^2 \alpha_{k-1}\beta^{-1}_{k-1}(\beta^{-1}_{k-1} - 1), \tag{4.34}$$

given the expected number of newborn targets, $\hat{N}_{b,k} = \langle 1, D_b\rangle$, and the expected number of survived targets, $\hat{N}_{s,k} = \langle p_s, D_{k-1}\rangle = \alpha_{k-1}\beta^{-1}_{k-1}\langle p_s, \varsigma\rangle$.

Note that (4.31) and (4.33) are essentially the same expressions as those found in the standard CPHD prediction. Also, similarly to the CPHD algorithm, the predictions of the intensity function and of the cardinality distribution are uncoupled. The main difference is in the way the predicted cardinality distribution is approximated, which for the DG-CPHD filter is modeled in terms of the discrete Gamma cardinality parameters $\alpha_{k|k-1}$ and $\beta_{k|k-1}$.

4.5.4 Discrete-Gamma CPHD Measurement Update

For the correction procedure, by knowing the prior intensity function, $D_{k|k-1}$, and the prior cardinality parameters, $\alpha_{k|k-1}$ and $\beta_{k|k-1}$, the measurement update step incorporates the set of all measurements taken at time instant k (realization Y_k) to calculate the posterior intensity function, D_k, and posterior cardinality parameters, α_k and β_k, which are functions of the posterior cardinality mean, $\mu_{N,k}$, and of the posterior cardinality variance, $\sigma^2_{N,k}$. The following proposition declares how the posterior intensity and cardinality parameters shall be calculated. Proofs are presented in the Section 4.8.

Proposition 4.2. *Suppose that the prior intensity function at time instant k, $D_{k|k-1}$, and the parameters of a prior discrete-Gamma cardinality distribution, $\alpha_{k|k-1}$ (shape) and $\beta_{k|k-1}$ (rate), are known. Given a set of m_k collected measurements that define a realization Y_k of the observation random finite set Ψ_k, then, under Assumptions 4.4–4.8, the posterior intensity*

function, D_k, and the posterior cardinality parameters, α_k and β_k, are given by

$$D_k(\mathrm{x}) \approx \frac{q_d(\mathrm{x})}{\alpha_{k|k-1}\beta_{k|k-1}^{-1}}\Theta_k\left[\mathrm{Y}_k\right]D_{k|k-1}(\mathrm{x}) + \frac{p_d(\mathrm{x})}{\alpha_{k|k-1}\beta_{k|k-1}^{-1}}\sum_{\mathrm{y}\in\mathrm{Y}_k}\frac{\ell_{\mathrm{y}}(\mathrm{x})}{\lambda c(\mathrm{y})}\Theta_k\left[\mathrm{Y}_k\setminus\{\mathrm{y}\}\right]D_{k|k-1}(\mathrm{x}), \tag{4.35}$$

$$\alpha_k \approx \frac{\mu_{N,k}^2}{\sigma_{N,k}^2},\ \beta_k \approx \frac{\mu_{N,k}}{\sigma_{N,k}^2}, \tag{4.36}$$

where

$$\mu_{N,k} = \theta_{1,0} + \theta_{0,1}\cdot\langle q_d,\varsigma\rangle, \tag{4.37}$$

$$\sigma_{N,k}^2 = \theta_{2,0} - \theta_{1,0} + 2\theta_{1,1}\cdot\langle q_d,\varsigma\rangle + \theta_{0,2}\cdot\langle q_d,\varsigma\rangle^2 - \mu_{N,k}^2 + \mu_{N,k}, \tag{4.38}$$

and where the following definitions apply:

$$\Theta_k\left[\mathrm{Y}\right] \triangleq \frac{\sum_{j=0}^{|\mathrm{Y}|}\hat{G}^{(j+1)}(\langle q_d,\varsigma\rangle)\bar{\sigma}_j(\mathrm{Y}\ )}{\sum_{i=0}^{m_k}\hat{G}^{(i)}\quad(\langle q_d,\varsigma\rangle)\bar{\sigma}_i(\mathrm{Y}_k)}, \tag{4.39}$$

$$\theta_{u,v} \triangleq \frac{\sum_{j=0}^{m_k} j^u\hat{G}^{(j+v)}(\langle q_d,\varsigma\rangle)\bar{\sigma}_j(\mathrm{Y}_k)}{\sum_{i=0}^{m_k}\quad\hat{G}^{(i)}\quad(\langle q_d,\varsigma\rangle)\bar{\sigma}_i(\mathrm{Y}_k)}, \tag{4.40}$$

$$\bar{\sigma}_i(\mathrm{Y}) \triangleq \sigma_{m=|\mathrm{Y}|,i}\left(\frac{\langle p_d\ell_{\mathrm{y}_1},\varsigma\rangle}{\lambda c(\mathrm{y}_1)},\ldots,\frac{\langle p_d\ell_{\mathrm{y}_m},\varsigma\rangle}{\lambda c(\mathrm{y}_m)}\right), \tag{4.41}$$

$$\hat{G}^{(\ell)}(x) = \frac{d^\ell}{dx^\ell}\mathrm{Li}_{-\alpha+1}(e^{-\beta}x), \tag{4.42}$$

for $\alpha = \alpha_{k|k-1}$, $\beta = \beta_{k|k-1}$ and $\ell \geq 1$. The function $\sigma_{m,i}(x_1,\ldots,x_m)$ is the elementary homogeneous symmetric function of degree i in $x_1,\ldots,x_m$.

Similarly to the standard CPHD filter, the posterior number of targets in the DG-CPHD recursion shall be estimated in the maximum-a-posteriori sense, i.e.,

$$\hat{N}_k \approx \arg\max_n p_k(n) = \arg\max_n\left(p_k(\lfloor n_m\rfloor), p_k(\lceil n_m\rceil)\right),\quad n_m = \frac{(\alpha_k-1)}{\beta_k}. \tag{4.43}$$

Note that (4.35) and (4.37) are analogous to those found in the standard CPHD correction step. In addition, as is the case of the CPHD recursion, the measurement updates of the intensity function and of the cardinality distribution are coupled. For the measurement update step, the main difference between the DG-CPHD and CPHD filters is in the way the cardinality distribution is approximated, which for the DG-CPHD filter is in terms of the discrete Gamma cardinality parameters α_k and β_k. It is important to mention that computing (4.42) is not a complex operation because, typically, $\mathrm{Li}_{-\alpha+1}(e^{-\beta}x)$ can be calculated to double floating-point precision with only few hundreds terms, and the pre-evaluated terms $(n^{\alpha-1}e^{-\beta n}x^n)_{n\in\mathbb{N}}$ that compose $\mathrm{Li}_{-\alpha+1}(e^{-\beta}x)$ can be iteratively updated by element-wise vector multiplications (or sums of logarithms) in order to evaluate the sequence $(\mathrm{Li}^{(\ell)}_{-\alpha+1}(e^{-\beta}x))_{\ell\in\mathbb{N}}$. To see this just

observe that

$$\begin{aligned}
\mathrm{Li}^{(1)}_{-\alpha+1}(e^{-\beta}x) &= x^{-1}\sum_{n=0}^{\infty} n^{\alpha-1}e^{-\beta n}x^n \cdot n,\\
\mathrm{Li}^{(2)}_{-\alpha+1}(e^{-\beta}x) &= x^{-2}\sum_{n=0}^{\infty} n^{\alpha-1}e^{-\beta n}x^n \cdot n(n-1),\\
&\vdots\\
\mathrm{Li}^{(\ell)}_{-\alpha+1}(e^{-\beta}x) &= x^{-\ell}\sum_{n=0}^{\infty} n^{\alpha-1}e^{-\beta n}x^n \cdot n\ldots(n-\ell+1),
\end{aligned} \tag{4.44}$$

whose terms can be iteratively computed as

$$\begin{aligned}
D^1_x(n^{\alpha-1}e^{-\beta n}x^n)_n &\coloneqq x^{-1}D^0_x(n^{\alpha-1}e^{-\beta n}x^n)_n \circ (n-0)_n,\\
D^2_x(n^{\alpha-1}e^{-\beta n}x^n)_n &\coloneqq x^{-1}D^1_x(n^{\alpha-1}e^{-\beta n}x^n)_n \circ (n-1)_n,\\
&\vdots
\end{aligned} \tag{4.45}$$

where the terms within parentheses represent sequences as $(\cdot)_{n\in\mathbb{N}}$, the symbol $\circ$ is the Hadamard product and D^{ℓ}_x is Euler's notation for differentiation.

4.5.5 Implementation of the DG-CPHD Filter via Gaussian Mixtures

As follows, the recursion established by *Propositions 4.1* and *4.2* will be presented in closed form by modeling the intensity functions as Gaussian mixtures. Without loss of generality in terms of applicability of the proposed filter, but in favor of illustration simplicity, we derive a solution for linear Gaussian models. In this context, the single-target state transition kernel and observation model are assumed linear and Gaussian as

$$p_t(\mathrm{x}|\mathrm{x}') = \mathcal{N}(\mathrm{x};\mathrm{F}\mathrm{x}',\mathrm{Q}), \tag{4.46}$$

$$\ell_{\mathrm{y}}(\mathrm{x}) \triangleq p_{\ell}(\mathrm{y}|\mathrm{x}\,) = \mathcal{N}(\mathrm{y};\mathrm{H}\mathrm{x}\,,\mathrm{R}). \tag{4.47}$$

Additionally we assume that the probabilities of survival and detection, p_s and p_d respectively, are independent of the state. The state point process (RFS) at a previous time instant $k-1$, and the target birth point process (RFS) at the time instant k are characterized by the following intensity models:

$$D_{k-1}(\mathrm{x}) = \sum_{i=1}^{\mathrm{I}_{k-1}} w^{(i)}_{k-1}\mathcal{N}(\mathrm{x};\mathrm{m}^{(i)}_{k-1},\mathrm{P}^{(i)}_{k-1}), \tag{4.48}$$

$$D_{b,k}\,(\mathrm{x}) = \sum_{i=1}^{\mathrm{I}_{b,k}} w^{(i)}_{b,k}\mathcal{N}(\mathrm{x};\mathrm{m}^{(i)}_{b,k},\mathrm{P}^{(i)}_{b,k}), \tag{4.49}$$

where $\{w^{(i)}_{k-1},\mathrm{m}^{(i)}_{k-1},\mathrm{P}^{(i)}_{k-1} : i=1,\ldots,\mathrm{I}_{k-1}\}$ is the set of weights, means and covariances that describe the state intensity function, and $\{w^{(i)}_{b,k},\mathrm{m}^{(i)}_{b,k},\mathrm{P}^{(i)}_{b,k} : i=1,\ldots,\mathrm{I}_{b,k}\}$ is the set of weights, means and covariances that describe the target birth intensity function. We shall not prove the

Gaussian-mixture DG-CPHD recursion formally, but rather refer the reader to [203] and just point out that, in essence, their proof is valid for the DG-CPHD recursion, except for a few different details.

Gaussian-Mixture DG-CPHD Prediction

The following equations form the prediction step for the Gaussian-mixture DG-CPHD recursion.

$$D_{k|k-1}(\mathrm{x}) = D_{b,k}(\mathrm{x}) + D_{s,k|k-1}(\mathrm{x}) \coloneqq \sum_{i=1}^{\mathrm{I}_{k|k-1}} w_{k|k-1}^{(i)} \mathcal{N}(\mathrm{x}; \mathrm{m}_{k|k-1}^{(i)}, \mathrm{P}_{k|k-1}^{(i)}), \tag{4.50}$$

$$\alpha_{k|k-1} \approx \frac{\mu_{N,k|k-1}^2}{\sigma_{N,k|k-1}^2},\ \beta_{k|k-1} \approx \frac{\mu_{N,k|k-1}}{\sigma_{N,k|k-1}^2}, \tag{4.51}$$

$$\mu_{N,k|k-1} \approx \sum_{i=1}^{\mathrm{I}_{b,k}} w_{b,k}^{(i)} + p_s \alpha_{k-1} \beta_{k-1}^{-1}, \tag{4.52}$$

$$\sigma_{N,k|k-1}^2 \approx \mu_{N,k|k-1} + p_s^2 \alpha_{k-1} \beta_{k-1}^{-1} \left(\beta_{k-1}^{-1} - 1\right), \tag{4.53}$$

where $\mathrm{I}_{k|k-1} = \mathrm{I}_{b,k} + \mathrm{I}_{k-1}$, and

$$\left\{ \begin{array}{c} w_{k|k-1}^{(i)}, \\ \mathrm{m}_{k|k-1}^{(i)}, \\ \mathrm{P}_{k|k-1}^{(i)} \end{array} \right\} \coloneqq \left\{ \begin{array}{ll} \left\{ \begin{array}{c} w_{b,k}^{(i)}, \\ \mathrm{m}_{b,k}^{(i)}, \\ \mathrm{P}_{b,k}^{(i)} \end{array} \right\}, & i \in [1..\mathrm{I}_{b,k}]; \\ \left\{ \begin{array}{c} p_s w_{k-1}^{(i)}, \\ \mathrm{F}\mathrm{m}_{k-1}^{(i)}, \\ \mathrm{F}\mathrm{P}_{k-1}^{(i)}\mathrm{F}^T + \mathrm{Q} \end{array} \right\}, & i \in]\mathrm{I}_{b,k}..\mathrm{I}_{k|k-1}]. \end{array} \right. \tag{4.54}$$

Gaussian-Mixture DG-CPHD Measurement Update

The equations in this section lay out the correction step for the Gaussian-mixture DG-CPHD recursion.

$$\begin{aligned} D_k(\mathrm{x}) &= D_{u,k}(\mathrm{x}) + D_{d,k}(\mathrm{x}) \\ &= \sum_{i=1}^{\mathrm{I}_{k|k-1}} w_{u,k}^{(i)} \mathcal{N}(\mathrm{x}; \mathrm{m}_{u,k}^{(i)}, \mathrm{P}_{u,k}^{(i)}) + \sum_{i=1}^{m_k \mathrm{I}_{k|k-1}} w_{d,k}^{(i)} \mathcal{N}(\mathrm{x}; \mathrm{m}_{d,k}^{(i)}, \mathrm{P}_{d,k}^{(i)}) \\ &\coloneqq \sum_{\ell=1}^{\mathrm{I}_k} w_k^{(\ell)} \mathcal{N}(\mathrm{x}; \mathrm{m}_k^{(\ell)}, \mathrm{P}_k^{(\ell)}), \end{aligned} \tag{4.55}$$

$$\alpha_k \approx \frac{\mu_{N,k}^2}{\sigma_{N,k}^2},\ \beta_k \approx \frac{\mu_{N,k}}{\sigma_{N,k}^2}, \tag{4.56}$$

$$\mu_{N,k} = \theta_{1,0} + \theta_{0,1} \cdot (1 - p_d), \tag{4.57}$$

$$\sigma_{N,k}^2 = \theta_{2,0} - \theta_{1,0} + 2\theta_{1,1} \cdot (1 - p_d) + \theta_{0,2} \cdot (1 - p_d)^2 - \mu_{N,k}^2 + \mu_{N,k}, \tag{4.58}$$

where $D_{u,k}$ is modeled as a Gaussian mixture describing the intensity of a random finite set composed of missed detections and $D_{d,k}$ is modeled as a Gaussian mixture describing the intensity of a random finite set composed of detected targets. For m_k collected measurements, $\mathrm{I}_k = \mathrm{I}_{k|k-1} + m_k \mathrm{I}_{k|k-1}$, where $\mathrm{I}_{k|k-1}$ components correspond to the missed detections and $m_k \mathrm{I}_{k|k-1}$ components correspond to the update of each prior component by each measurement. The missed detection components are evaluated for $i = 1, \ldots, \mathrm{I}_{k|k-1}$ as

$$w_{u,k}^{(i)} = \frac{(1-p_d)\Theta_k\left[\mathrm{Y}_k\right]}{\alpha_{k|k-1}\beta_{k|k-1}^{-1}} w_{k|k-1}^{(i)}, \tag{4.59}$$

$$\mathrm{m}_{u,k}^{(i)} = \mathrm{m}_{k|k-1}^{(i)}, \tag{4.60}$$

$$\mathrm{P}_{u,k}^{(i)} = \mathrm{P}_{k|k-1}^{(i)}. \tag{4.61}$$

For each measurement $\mathrm{y} \in \mathrm{Y}_k$, for all $i \in [1..\mathrm{I}_{k|k-1}]$, the detected components are given by

$$w_{d,k}^{(i)}(\mathrm{y}) = \frac{\Theta_k\left[\mathrm{Y}_k \setminus \{\mathrm{y}\}\right] p_d\, \hat{\ell}_{k,\mathrm{y}}^{(i)}}{\alpha_{k|k-1}\beta_{k|k-1}^{-1}\lambda c(\mathrm{y})} w_{k|k-1}^{(i)}, \tag{4.62}$$

$$\mathrm{m}_{d,k}^{(i)}(\mathrm{y}) = \mathrm{m}_{k|k-1}^{(i)} + \mathrm{K}_k^{(i)}(\mathrm{y} - \mathrm{Hm}_{k|k-1}^{(i)}), \tag{4.63}$$

$$\mathrm{P}_{d,k}^{(i)}(z) = (\mathbb{I}_{d_x} - \mathrm{K}_k^{(i)}\mathrm{H})\mathrm{P}_{k|k-1}^{(i)}, \tag{4.64}$$

$$\begin{aligned}
\hat{\ell}_{k,\mathrm{y}}^{(i)} &= \mathcal{N}(\mathrm{y}; \mathrm{Hm}_{k|k-1}^{(i)}, \mathrm{S}_{k|k-1}^{(i)}), \\
\mathrm{K}_k^{(i)} &= \mathrm{P}_{k|k-1}^{(i)}\mathrm{H}^T\mathrm{S}_{k|k-1}^{(i)-1}, \\
\mathrm{S}_{k|k-1}^{(i)} &= \mathrm{HP}_{k|k-1}^{(i)}\mathrm{H}^T + \mathrm{R}.
\end{aligned}$$

Thus, the posterior components in (4.55) are set as

$$\left\{ \begin{array}{c} w_k^{(j\cdot \mathrm{I}_{k|k-1}+i)}, \\ \mathrm{m}_k^{(j\cdot \mathrm{I}_{k|k-1}+i)}, \\ \mathrm{P}_k^{(j\cdot \mathrm{I}_{k|k-1}+i)} \end{array} \right\} := \left\{ \begin{array}{ll} \left\{ \begin{array}{c} w_{u,k}^{(i)}, \\ \mathrm{m}_{u,k}^{(i)}, \\ \mathrm{P}_{u,k}^{(i)} \end{array} \right\}, & i \in [1..\mathrm{I}_{k|k-1}], \\ & j = 0; \\ \left\{ \begin{array}{c} w_{d,k}^{(i)}(\mathrm{y}_j), \\ \mathrm{m}_{d,k}^{(i)}(\mathrm{y}_j), \\ \mathrm{P}_{d,k}^{(i)}(\mathrm{y}_j) \end{array} \right\}, & i \in [1..\mathrm{I}_{k|k-1}], \\ & j \in [1..m_k]. \end{array} \right. \tag{4.65}$$

The Gaussian-Mixture DG-CPHD filter is presented in *Algorithm 4.2*. Theoretically, the filter requires a geometrically increasing number of Gaussian components as time progresses in a similar way as the Gaussian-Mixture PHD filter. Hence, practical implementations require techniques to keep the total number of components at a maximum, such as pruning and merging components as described in [201]. In this procedure, components with negligible weights are pruned and components that are close together are merged. Tracks are extracted as per the extraction algorithm presented in [201].

4.5.6 Algorithm complexity and implementation details

In the context of the original CPHD filter, the attempt of propagating a nowhere vanishing cardinality distribution, $p_N(n)$, would be intractable since this would involve estimating an

infinite number of terms, for $n \in \mathbb{N}_0$. However, as touched on in [129], it is generally safe to assume that, in practical problems, cardinality distributions are short or moderate tailed, which allows them to be truncated at some number $n = n_{\max} \geq \nu$. This corresponds to say that their probability generating functions are polynomials of degrees not exceeding ν. Therefore, when talking about complexity of the CPHD filter, the usual argument observes that, by assuming the prior cardinality p.g.f. to be a polynomial of degree not exceeding ν, i.e., $\deg G_{k|k-1}(x) \leq \nu$, then $\deg G_k(x) \leq \nu$, and in the next filtering step $\deg G_{k+1|k}(x) \leq \nu + \deg G_{b,k+1}(x)$ [129]. The conclusion of this argument is that increasing the possible number of targets in the scene will not directly affect the computational requirements for cardinality.

According to the analysis provided in [129] and [203], most of the CPHD computational effort is due to $m+1$ evaluations of sets of elementary symmetric functions, each computed with effort $\mathcal{O}(m^2)$, which would render a complexity bounded by $\mathcal{O}(m^3)$ operations, for m measurements. This complexity can be reduced to $\mathcal{O}(m^2 \log^2 m)$ for a procedure mentioned in [203]. Nevertheless, such analysis emphasizes that evaluations of elementary symmetric functions will dominate the complexity for a high number of measurements, and so disregards about $(n_{\max}m - m^2/2 + n_{\max} - m)(m+1) \sim \mathcal{O}(n_{\max}m^2)$ calculations[6] for $n_{\max} > m$, involving multiple-term multiplications[7] that are necessary to compute $\Upsilon_k[\mathrm{Y}_k]$, $\Upsilon_k[\mathrm{Y}_k \setminus \{\mathrm{y}\}]$ and $p_k(n)$, according to equations (4.15) and (4.18) presented in Section 4.3.2.

Not less important, although not usually taken into account, is observing that as the number of targets increases, $n_{\max}$ must be set accordingly to properly describe probability masses that become increasingly important towards higher values of n, while maintaining accuracy of the cardinality representation. How much $n_{\max}$ should increase depends on how much information measurements provide about the number of targets, which ultimately depends on the signal-to-noise ratio.

Proposition 4.3. *Considering a constant probability of detection $p_d(x) = p_d$, Poisson-distributed false alarms at rate λ, and setting $\varepsilon \coloneqq \Pr\{0 < n \leq n_{max}\} \in (0,1)$, it follows that*

$$\begin{aligned} n_{max} &\geq \frac{2\mu_{N,k}}{\varepsilon}\left(1 + \sqrt{(1-\varepsilon) - \varepsilon\left(\frac{\sigma_{N,k}}{\mu_{N,k}}\right)^2}\right) \\ &\geq K \cdot \mu_{N,k|k-1}, \end{aligned} \tag{4.66}$$

for $K \geq 1$ and where $0 < \sigma^2_{N,k} \leq \mu_{N,k} n_{max} - \mu^2_{N,k}$ is assumed.

The point we wish to make is that $K \cdot \mu_{N,k|k-1} > m$ is not rare (e.g., high number of targets and low number of false alarms) and, in that case, the CPHD algorithm complexity should be bounded by approximately $m^3 + n_{\max}m^2 - m^2/2 \sim \mathcal{O}(n_{\max}m^2)$ operations since $n_{\max} > m$. The proof of *Proposition 4.3* is presented in the Section 4.8. In general, as $\lambda \to 0$ the CPHD algorithm complexity is rather dominated by the cardinality parameterization and by the birth model, and becomes increasingly complex as the number of target-generated measurements increases. This fact corroborates Mahler's perception that further approximation of the CPHD filter may enable a computational gain [130].

In the DG-CPHD algorithm context, the computational complexity is alleviated by the fact that, apart from the $\mathcal{O}(m^3)$ operations for the elementary symmetric functions, only about

[6] Or $(n_{\max}m/2 + n_{\max} + 1)(n_{\max} + 1) \sim \mathcal{O}(n^2_{\max}m)$ for $n_{\max} < m$.
[7] Exponentiating sums of logarithms and multiplying by one elementary symmetric function.

$\bar{n}_{\max}(m+3) \sim \mathcal{O}(\bar{n}_{\max} m)$ operations are required for evaluating derivatives of the prior cardinality p.g.f.. In this context, $\bar{n}_{\max}$ is the number of terms used to approximate each infinite sum in (4.44), and about $6m + m(m-1)$ operations are needed for the terms that sum up to give $\Theta_k[\mathrm{Y}_k]$, $\Theta_k[\mathrm{Y}_k \setminus \{\mathrm{y}\}]$ and $\theta_{u,v}$, according to equations (4.39) and (4.40) in Section 4.5.4. The overall complexity of the DG-CPHD algorithm is then bounded by approximately $m^3 + \bar{n}_{\max} m + m^2 \sim \mathcal{O}(m^3)$ operations, where $\bar{n}_{\max}$ must be set to meet an adequate accuracy for evaluating the prior cardinality p.g.fl. but has loose relationship with the number of measurements. The scheme to compute $\mathrm{Li}^{(\ell)}_{-\alpha+1}(e^{-\beta}x)$ for $\ell \in [0..m+2]$, with approximately $\bar{n}_{\max}(m+3)$ operations, evaluates $\bar{n}_{\max}$ log-terms of $\mathrm{Li}^{(0)}_{-\alpha+1}(e^{-\beta}x)$, and then applies an iterative procedure over $m+2$ steps, each calculating $\bar{n}_{\max}$ log-increments to update the log-terms in view of (4.45). This procedure is made explicit in *Algorithm 4.1.*

Algorithm 4.1: Normalized Derivatives of Polylogarithms

Input : α, β, ζ, ε (machine precision), m

$l_\zeta \coloneqq \log \zeta$, $l_z \coloneqq -\beta + l_\zeta$

Compute bounds for number of terms

$\bar{n}_{\min} \coloneqq 1$, $\bar{n}_0 \coloneqq \hat{N}_{k|k-1}$

/* Find $\bar{n}_{\max}$ that corresponds to the smallest representable term $n^{\alpha-1}e^{-\beta n}$ by Newton's method */

for $i = 1, \ldots, N_{iter}$ **do**

  $\bar{n}_i \leftarrow \bar{n}_{i-1} - \frac{(\alpha-1)\log \bar{n}_{i-1} - \beta \bar{n}_{i-1} - \log\, \varepsilon}{2((\alpha-1)/\bar{n}_{i-1} - \beta)}$

end

$\bar{n}_{\max} \coloneqq \bar{n}_i$

$(n)_n = (n)_{n \in [\bar{n}_{\min} .. \bar{n}_{\max}]} \coloneqq (\bar{n}_{\min}, \bar{n}_{\min}+1, \ldots, \bar{n}_{\max})$

$(nl_z)_n \coloneqq (n)_n \cdot l_z$

$(\log n)_n \coloneqq \log(n)_n$

$((\alpha-1)\log n)_n \coloneqq (\alpha-1) \cdot (\log n)_n$

$(\vartheta(n))_n \coloneqq ((\alpha-1)\log n)_n + (nl_z)_n$

$\vartheta_{\max} \coloneqq \max(\vartheta(n))_n$

Evaluate polynomials

$L(0) \coloneqq \sum_{n=1}^{n_{\max}} \exp((\vartheta(n))_n - \vartheta_{\max})$

$l_{\zeta,1} \coloneqq l_\zeta$

$(\log(n^{\underline{1}}))_n \coloneqq (\log n)_n$

for $\ell = 1, \ldots, m+2$ **do**

  $L(\ell) \coloneqq \sum_{n=1}^{n_{\max}} \exp((\vartheta(n))_n - \vartheta_{\max} + (\log(n^{\underline{\ell}}))_n - l_{\zeta,\ell})$

  $(\log(n^{\underline{\ell+1}}))_n \leftarrow (\log(n^{\underline{\ell}}))_n + (\log(n-\ell))_n$

  $l_{\zeta,\ell+1} \leftarrow l_{\zeta,\ell} + l_\zeta$

end

Output: Approximations of higher-order derivatives

$$(\mathrm{Li}^{(\ell)}_{1-\alpha}(e^{-\beta}\zeta))_{\ell \in [0..m+2]} \propto \frac{(L(\ell))_{\ell \in [0..m+2]}}{\max \left| (L(\ell))_{\ell \in [0..m+2]} \right|}$$

Algorithm 4.2: Gaussian-Mixture DG-CPHD Filter

Input : $\{w_{k-1}^{(i)}, \mathrm{m}_{k-1}^{(i)}, \mathrm{P}_{k-1}^{(i)}\}_{i\in[1..\mathrm{I}_{k-1}]}$, α_{k-1}, β_{k-1}, $\mathrm{Y}_k = \{\mathrm{y}_1, \ldots, \mathrm{y}_{m_k}\}$

Prediction Step

/* Prediction of newborn target intensity */

for $i = 1, \ldots, \mathrm{I}_{b,k}$ **do**

  $w_{k|k-1}^{(i)} \coloneqq w_{b,k}^{(i)}$, $\mathrm{m}_{k|k-1}^{(i)} \coloneqq \mathrm{m}_{b,k}^{(i)}$, $\mathrm{P}_{k|k-1}^{(i)} \coloneqq \mathrm{P}_{b,k}^{(i)}$

end

/* Prediction of surviving target intensity */

$\mathrm{I}_{k|k-1} \coloneqq \mathrm{I}_{b,k} + \mathrm{I}_{k-1}$

for $i = \mathrm{I}_{b,k} + 1, \ldots, \mathrm{I}_{k|k-1}$ **do**

  $w_{k|k-1}^{(i)} \coloneqq p_s w_{k-1}^{(i)}$, $\mathrm{m}_{k|k-1}^{(i)} \coloneqq \mathrm{F}\mathrm{m}_{k-1}^{(i)}$, $\mathrm{P}_{k|k-1}^{(i)} \coloneqq \mathrm{F}\mathrm{P}_{k-1}^{(i)}\mathrm{F}^T + \mathrm{Q}$

end

/* Prediction of cardinality parameters */

$\mu_{N,k|k-1} \coloneqq \sum_{i=1}^{\mathrm{I}_{b,k}} w_{b,k}^{(i)} + p_s\alpha_{k-1}\beta_{k-1}^{-1}$, $\sigma_{N,k|k-1}^2 \coloneqq \mu_{N,k|k-1} + p_s^2\alpha_{k-1}\beta_{k-1}^{-1}(\beta_{k-1}^{-1} - 1)$

$\alpha_{k|k-1} \coloneqq \left\lfloor \mu_{N,k|k-1}^2/\sigma_{N,k|k-1}^2 \right\rfloor$, $\beta_{k|k-1} \coloneqq \mu_{N,k|k-1}/\sigma_{N,k|k-1}^2$

Measurement Update Step

/* Pre-computations for updated components */

for $i = 1, \ldots, \mathrm{I}_{k|k-1}$ **do**

  $\bar{\mathrm{y}}_{k|k-1}^{(i)} \coloneqq \mathrm{H}\mathrm{m}_{k|k-1}^{(i)}$, $\mathrm{S}_{k|k-1}^{(i)} \coloneqq \mathrm{H}\mathrm{P}_{k|k-1}^{(i)}\mathrm{H}^T + \mathrm{R}$,

  $\mathrm{K}_k^{(i)} \coloneqq \mathrm{P}_{k|k-1}^{(i)}\mathrm{H}^T\mathrm{S}_{k|k-1}^{(i)-1}$, $\mathrm{P}_k^{(i)} \coloneqq (\mathbb{I}_{d_x} - \mathrm{K}_k^{(i)}\mathrm{H})\mathrm{P}_{k|k-1}^{(i)}$

end

$\{\hat{\ell}_{k,\mathrm{y}}^{(i)}\} \triangleq \{\mathcal{N}(\mathrm{y}; \bar{\mathrm{y}}_{k|k-1}^{(i)}, \mathrm{S}_{k|k-1}^{(i)})\}_{i\in[1..\mathrm{I}_{k|k-1}],\mathrm{y}\in\mathrm{Y}_k}$

Obtain $\{\bar{\sigma}_j(\mathrm{Y})\}_{j=1}^{|\mathrm{Y}|}$ (4.41), for $\mathrm{Y} = \mathrm{Y}_k, \mathrm{Y}_k \setminus \{\mathrm{y}_1\}, \ldots \mathrm{Y}_k \setminus \{\mathrm{y}_{m_k}\}$, and $\langle q_d\ell_{\mathrm{y}_j}, \varsigma\rangle = \alpha_{k|k-1}^{-1}\beta_{k|k-1}q_d\sum_{i=1}^{\mathrm{I}_{k|k-1}} w_{k|k-1}^{(i)}\hat{\ell}_{k,\mathrm{y}_j}^{(i)}$

Compute $(\mathrm{Li}_{1-\alpha_{k|k-1}}^{(\ell)}(e^{-\beta_{k|k-1}}q_d))_{\ell\in[0..m_k+2]}$ (*Algorithm 4.1*)

Evaluate $\theta_{0,1} \equiv \Theta_k[\mathrm{Y}_k]$, $\theta_{0,2}$, $\theta_{1,0}$, $\theta_{2,0}$, $\theta_{1,1}$ (4.40)

/* Update of missed-detection intensity */

for $i = 1, \ldots, \mathrm{I}_{k|k-1}$ **do**

  $w_k^{(i)} \coloneqq q_d\,\alpha_{k|k-1}^{-1}\beta_{k|k-1}\Theta_k[\mathrm{Y}_k]\,w_{k|k-1}^{(i)}$, $\mathrm{m}_k^{(i)} \coloneqq \mathrm{m}_{k|k-1}^{(i)}$, $\mathrm{P}_k^{(i)} \coloneqq \mathrm{P}_{k|k-1}^{(i)}$

end

/* Update of detected target intensity */

$\mathrm{I}_k \coloneqq (1 + m_k)\mathrm{I}_{k|k-1}$

for $j = 1, \ldots, m_k$ **do**

  Evaluate $\Theta_k[\mathrm{Y}_k \setminus \{\mathrm{y}_j\}]$ (4.39)

  for $i = 1, \ldots, \mathrm{I}_{k|k-1}$ **do**

    $w_k^{(j\cdot\mathrm{I}_{k|k-1}+i)} \coloneqq \frac{\Theta_k[\mathrm{Y}_k\setminus\{\mathrm{y}_j\}]p_d\,\hat{\ell}_{k,\mathrm{y}_j}^{(i)}w_{k|k-1}^{(i)}}{\alpha_{k|k-1}\beta_{k|k-1}^{-1}\gamma c(\mathrm{y}_j)}$,

    $\mathrm{m}_k^{(j\cdot\mathrm{I}_{k|k-1}+i)} \coloneqq \mathrm{m}_{k|k-1}^{(i)} + \mathrm{K}_k^{(i)}(\mathrm{y}_j - \mathrm{H}\mathrm{m}_{k|k-1}^{(i)})$, $\mathrm{P}_k^{(j\cdot\mathrm{I}_{k|k-1}+i)} \coloneqq \mathrm{P}_k^{(i)}$

  end

end

/* Update of cardinality parameters */

$\mu_{N,k} \coloneqq \theta_{1,0} + \theta_{0,1}\cdot q_d$, $\sigma_{N,k}^2 \coloneqq \theta_{2,0} - \theta_{1,0} + 2\theta_{1,1}\cdot q_d + \theta_{0,2}\cdot q_d^2 - \mu_{N,k}^2 + \mu_{N,k}$

$\alpha_k \coloneqq \left\lfloor \mu_{N,k}^2/\sigma_{N,k}^2 \right\rfloor$, $\beta_k \coloneqq \mu_{N,k}/\sigma_{N,k}^2$

Output: $\{w_k^{(i)}, \mathrm{m}_k^{(i)}, \mathrm{P}_k^{(i)}\}_{i\in[1..\mathrm{I}_k]}$, α_k, β_k

4.6 Numerical experiment

In this section, we present experimental results for a simple example, very similar to the examples presented in [203], but modulating difficulty by varying either the number of targets that appear, the average number of false alarms per frame, or the probability of detection. The intent is to show differences of performance and computational effort of the PHD, CPHD, and DG-CPHD filters in difficult situations, where we hope to demonstrate benefits of the approximations introduced by the DG-CPHD filter.

The example consists of a two-dimensional scenario where a number of targets, unknown a priori, may appear and are observed in the region $[-1000, +1000]\times[-1000, +1000]$ (m×m), with the possibility of false alarms. Each target is described by its state vector $\mathrm{x} = (p_x, p_y, v_x, v_y)^T$, where $\mathrm{p} = (p_x, p_y)^T$ is a pair that specifies a position in Cartesian coordinates and $\mathrm{v} = (v_x, v_y)^T$ is the pair specifying velocity in the same coordinates. At each time instant k, the state is written as $\mathrm{x}_k = \mathrm{x}(t_k)$. Each target is assumed to move with nearly-constant velocity, with transition matrix and state process covariance matrix given respectively by

$$\mathrm{F} = \begin{pmatrix} \mathbb{I}_2 & \mathbb{I}_2\Delta t \\ 0_2 & \mathbb{I}_2 \end{pmatrix}, \; \mathrm{Q} = \begin{pmatrix} \mathbb{I}_2\Delta t^3/3 & \mathbb{I}_2\Delta t^2/2 \\ \mathbb{I}_2\Delta t^2/2 & \mathbb{I}_2\Delta t \end{pmatrix}\sigma_q^2,$$

where $\mathbb{I}_2$ and 0_2 are the identity and zero matrices with dimensions 2×2, respectively, $\Delta t = 1\,\mathrm{s}$ is the sampling period, and the standard deviation of velocity increments is given by $\sigma_q = 5\,\mathrm{m/s}^{\frac{3}{2}}$. Each target remains in the scene up to the next time step with probability $p_s = 0.99$. A single sensor collects position measurements in the Cartesian space, corrupted by a Gaussian-distributed noise, characterized by the output matrix and measurement noise covariance matrix, respectively,

$$\mathrm{H} = \begin{pmatrix} \mathbb{I}_2 & 0_2 \end{pmatrix}, \; \mathrm{R} = \mathbb{I}_2\sigma_r^2,$$

where $\sigma_r = 10\,\mathrm{m}$ is the standard deviation of the measured positions. False alarms can be generated according to a Poisson point-process with intensity $D_c(\mathrm{y}) = \lambda \cdot c(\mathrm{y})$, where λ is the average number of false alarms per scan, and $c(\mathrm{y})$ is the spatial distribution of clutter, assumed uniform in the surveillance region with "volume" (area) $V = 2000^2\,\mathrm{m}^2$.

Each instance of the example is simulated for $T = 100\,\mathrm{s}$. Denote N_t as the total number of targets that appear and will remain till the end of a simulated instance, at $t = 100\,\mathrm{s}$. In all cases, targets appear in batches at positions uniformly sampled in the area $[-800, +800]\times[-800, +800]$ (m × m), and with random velocities uniformly sampled in the ranges $[-10, +10] \times [-10, +10]$ (m/s × m/s). The batches of target appearances are set as follows:

- $0.25N_t$ targets are already in the scene at $t = 0$ and will remain up to $t = 100\,\mathrm{s}$ with exception of 5 targets that are set to disappear at $t = 80\,\mathrm{s}$,
- $0.25N_t$ targets along with 2 other targets appear at $t = 20\,\mathrm{s}$ and are set to remain to the end,
- $0.25N_t$ of targets along with another target appear at $t = 40\,\mathrm{s}$ and are set to remain,
- $0.25N_t$ of targets along with 2 other targets appear at $t = 60\,\mathrm{s}$ and are set to remain,
- from $N_t + 5$ targets in the scene, 5 targets disappear at $t = 80\,\mathrm{s}$ and the remaining N_t stay up to $t = 100\,\mathrm{s}$.

The birth random finite set is assumed as a Poisson point-process with approximate intensity $D_b(\mathrm{x}) = \sum_{i=1}^{4} w_b^{(i)} \mathcal{N}(\mathrm{x}; \mathrm{m}_b^{(i)}, \mathrm{P}_b^{(i)})$, where

$$\begin{aligned}
\mathrm{m}_b^{(1)} &= (-500, -500, 0, 0)^T, \\
\mathrm{m}_b^{(2)} &= (-500, +500, 0, 0)^T, \\
\mathrm{m}_b^{(3)} &= (+500, -500, 0, 0)^T, \\
\mathrm{m}_b^{(4)} &= (+500, +500, 0, 0)^T,
\end{aligned}$$

and

$$w_b^{(i)} = \frac{N_t + 5}{4T/\Delta t} \,(\text{target/frame}),$$

$$\mathrm{P}_b^{(i)} = \begin{pmatrix} 500^2 \mathbb{I}_2 & 0_2 \\ 0_2 & 10^2 \mathbb{I}_2 \end{pmatrix}, \; i = 1, \ldots, 4.$$

For each filter, at each time step, pruning is performed based on a weight threshold of $\tau_{\mathrm{prn}} = 10^{-5}$ and merging with threshold of $\tau_{\mathrm{mrg}} = 4\,\mathrm{m}$, and the number of maintained Gaussian components is limited at $\mathrm{J}_{\max} = 100$ (see [201] for details on the pruning and merging procedure). Measurements are gated with gate-size probability of $p_{\mathrm{gate}} = 0.999$. For the CPHD and DG-CPHD filters, the estimated (posterior) number of targets is obtained in the maximum-a-posteriori sense. The cardinality distribution of the CPHD is estimated to a maximum of $n_{\max} = 2 \times N_t$ terms. This maximum number of cardinality terms has been chosen to keep the CPHD filter computational effort competitive in relation to the other filters for difficult scenarios.

We examine the DG-CPHD filter in comparison with the PHD and CPHD filters for three different cases as follows.

Case 1: Fixed probability of detection, $p_d = 0.98$, and fixed number of false alarms per scan, $\lambda = 50$, all filters are tested for different numbers of targets that appear and remain in the scene, $N_t \in \{10, 20, 30, 40, 50\}$.

Case 2: For fixed number of targets that appear and remain in the scene, $N_t = 20$, and fixed number of false alarms per scan, $\lambda = 50$, all filters are tested for different probabilities of detection, $p_d \in \{0.98, 0.90, 0.80, 0.70, 0.60\}$.

Case 3: For fixed number of targets that appear and remain in the scene, $N_t = 20$, and fixed probability of detection, $p_d = 0.80$, all filters are tested for different numbers of false alarms per frame, $\lambda \in \{10, 30, 50, 100, 200\}$.

For each case, 200 Monte Carlo (MC) runs are performed, each with an independently sampled set of trajectories (ground truth), independently generated clutter, and independently generated (target-originated) measurements for each trial. For all filters, performance is evaluated in terms of:

- mean Optimal Subpattern Assignment (OSPA) metric [174] for cut-off $c_{\mathrm{OSPA}} = 100$ and norm order $p_{\mathrm{OSPA}} = 1$,
- root-mean-square error (RMSE) of the estimated number of targets,
- estimated cardinality variance, and

- computation time (per time step).

All indexes are averaged over time steps and consolidated for all values of the varying parameters. In our comparisons, we use implementations of the PHD and CPHD filters in Matlab language retrieved from Ba-Tuong Vo's webpage[8], and the DG-CPHD filter code was also written in Matlab with the same structure. Note that the CPHD algorithm implementation available in Ba-Tuong Vo's web page (according to [203]) could be made more efficient and robust by computing all combinatorial coefficients just once and by normalizing them to avoid arithmetic overflow in extreme scenarios (e.g., $\lambda = 250$ false alarms per frame).

4.6.1 Results

Case 1

Figure 4.2 shows the tracks generated by the CPHD and DG-CPHD filters for an exemplar run of the first case, in a subcase where $p_d = 0.98$, $\lambda = 50$ and $N_t = 50$. For the same subcase, Figures 4.3 and 4.4 present, respectively, the mean OSPA, and cardinality mean and standard deviation over time for the PHD, CPHD, and DG-CPHD filters, where we can perceive the advantage of estimating the cardinality distribution.

The consolidated performance indexes for Case 1, averaged over all time steps for each $N_t \in \{10, 20, 30, 40, 50\}$, can be found in Figures 4.5, 4.6, 4.7, and 4.8. In these figures it becomes clear that, in this case, the performance of the DG-CPHD filter is very similar to that of the CPHD filter, but at a smaller computational cost, with practically the same computation times as that of the PHD filter. In terms of average mean OSPA metric the DG-CPHD filter shows a performance slightly better than that of the CPHD filter. For all filters, the average mean OSPA metric decreases for $N_t < 40$ and then increases at a small rate for $N_t \geq 40$. It is interesting to note that the average cardinality RMSE and variance of all filters seem to increase subexponentially (at a small rate) with the possible number of targets in the scene.

Case 2

For Case 2, in a subcase where $p_d = 0.60$, $\lambda = 50$ and $N_t = 20$, Figures 4.9 and 4.10 show, respectively, the mean OSPA, and cardinality mean and standard deviation over time for the PHD, CPHD and DG-CPHD filters. This is a difficult scenario, where it is clear the superiority of the CPHD and DG-CPHD filters in terms of the mean OSPA and cardinality statistics.

The consolidated performance indexes in Case 2, for each $p_d \in \{0.98, 0.90, 0.80, 0.70, 0.60\}$, averaged over time, can be found in Figures 4.11, 4.12, 4.13, and 4.14. In this case, the performance of the DG-CPHD filter is very similar to that of the CPHD filter in terms of average mean OSPA and cardinality variance, but slightly worse in terms of average cardinality RMSE. However, the computational cost of the DG-CPHD filter is much smaller than that of the CPHD filter, being rather comparable to that of the PHD filter. As expected, the error indexes and cardinality variance decrease as the probability of detection increases, for all filters.

[8] http://ba-tuong.vo-au.com/codes.html.

Case 3

For Case 3, in a very challenging subcase where $p_d = 0.80$, $\lambda = 200$ and $N_t = 20$, Figures 4.15 and 4.16 depict, respectively, the mean OSPA, and cardinality mean and standard deviation over time for the PHD, CPHD and DG-CPHD filters. Although the scenario is difficult for any filter, it is evident that the CPHD and DG-CPHD filters perform better than the PHD filter, being able to identify and track a greater number of targets on average. The CPHD filter is the best filter in this case, identifying more targets on average than the DG-CPHD filter. This is due to the cardinality bias introduced by the approximations adopted for the DG-CPHD filter.

Figures 4.17, 4.19, and 4.20 present the consolidated performance indexes for Case 3, averaged over time for each $\lambda \in \{10, 30, 50, 100, 200\}$. From the figures it remains no doubt that the performance of the DG-CPHD filter is very close to that of the CPHD filter for $\lambda \leq 100$, but less responsive to identify targets when the number of false alarms is sufficiently high. The cardinality RMSE of the DG-CPHD filter is greater than that of the CPHD filter, which is a direct consequence of the approximations introduced for evaluating the DG-CPHD cardinality moments. There is no doubt that the DG-CPHD filter requires less computational effort than the standard CPHD filter implementation, presenting computation times that are comparable to the PHD filter for this case. However, as can be readily seen in Figure 4.20, as λ increases the complexity of both the CPHD and DG-CPHD filters are dominated by the total number of measurements, i.e., the computation times increase subexponentially ($\mathcal{O}(m^3)$) with the number of false alarms. It is also worth noting that the average cardinality RMSE of all filters seem to increase subexponentially with the number of false alarms per frame.

4.7 Conclusions

All in all, this chapter addresses the fundamental problem of reducing computational complexity in multi-object estimation, which figures as one of the most important concerns of this dissertation. Specifically, the problem is dealt with by proposing a filter based on a very effective simplification of CPHD filter. The new filter is second-order in target number, where the targets' state is assumed to follow an independent and identically distributed cluster process with the cardinality distribution modeled as a discretized Gamma distribution. Our work capitalizes on Ronald Mahler's perception that one more step towards simplification of the CPHD algorithm implementation might be interesting. The strategy employed was to mimic, based on a discrete-Gamma distribution, the procedure of a Kalman filter for the cardinality random variable, i.e., estimating sufficient statistics. As demonstrated by the numerical examples, the discrete Gamma distribution allows simple calculations for approximating the first- and second-order moments of the posterior cardinality distribution, and efficiently addresses tracking scenarios with underdispersed and slightly overdispersed target counts, without the restrictions required by the binomial filter.

The results also demonstrate that the DG-CPHD filter is more computationally efficient than the CPHD filter implementation proposed in [203], especially for scenarios where a high number of CPHD cardinality terms, $n_{\max}$, is necessary, i.e., in situations where the number of target-generated measurements is significantly increased as many targets appear in a scene. The experimental results support our argument that the computational effort is dominated by

$\mathcal{O}(n_{\max}m^2)$ in this case, where $n_{\max} > m$ when the CPHD filter is more computationally complex than the DG-CPHD filter. In contrast, the DG-CPHD filter must rely on a finite number of terms, $\bar{n}_{\max}$, to approximate derivatives of the prior cardinality p.g.f., $\hat{G}^{(\ell)}_{k|k-1}(\langle q_d, \varsigma\rangle)$, and as the results show, $\bar{n}_{\max}$ is much less sensitive to the number of target-generated measurements. To conclude, both our complexity analysis and the experimental results suggest that the DG-CPHD algorithm has computational complexity somewhere between that of the PHD filter and that of the CPHD filter but cardinality accuracy and variance similar to that of the CPHD filter in its standard setting [203].

We conjecture that the procedure proposed in this chapter could be extended to other cardinality distributions of interest and might be the basis for future work.

4.8 Proofs

4.8.1 Proof of Proposition 4.1

Proof of Proposition 4.1. Consider the p.g.fl. of the prior state RFS,

$$\begin{aligned} G_{k|k-1}[h] &= \int h^{\mathrm{X}} p_{\Xi_{k|k-1}}(\mathrm{X}|\mathrm{Y}_{1:k-1})\,\delta\mathrm{X} \\ &= \int h^{\mathrm{X}} \left\{ \int p_{\Xi|\Xi_{k-1}}(\mathrm{X}|\mathrm{X}')\, p_{\Xi_{k-1}}(\mathrm{X}'|\mathrm{Y}_{1:k-1})\,\delta\mathrm{X}' \right\} \delta\mathrm{X} \\ &= \int G_t[h|\mathrm{X}'] p_{\Xi_{k-1}}(\mathrm{X}'|\mathrm{Y}_{1:k-1})\,\delta\mathrm{X}', \end{aligned}$$

where $G_t[h|\mathrm{X}']$ is the p.g.fl. of the multi-target transition kernel. Because targets are assumed to move independently (*Assumption 4.1*) and new targets are born independently (*Assumption 4.3*), we know that $G_t[h|\mathrm{X}'] = G_b[h] \cdot \prod_{i=1}^{n} G_t[h|\mathrm{x}'_i]$ where $G_t[h|\mathrm{x}'] = q_s(\mathrm{x}') + p_s(\mathrm{x}')p_{t,h}(\mathrm{x}')$ (using *Assumption 4.2*), with $p_{t,h}(\mathrm{x}') \triangleq \int_{\mathcal{X}} h(\mathrm{x})p_t(\mathrm{x}|\mathrm{x}')d\mathrm{x}$, and $p_t(\cdot|\mathrm{x}')$ is the single-target transition kernel. Since the birth RFS is assumed to be a Poisson point process (*Assumption 3*) with mean $\mu_b = N_b$ and intensity $D_b(\mathrm{x}) = \mu_b b(\mathrm{x})$:

$$\begin{aligned} G_{k|k-1}[h] &= G_b[h] \int (q_s + p_s p_{t,h})^{\mathrm{X}'} p_{\Xi_{k-1}}(\mathrm{X}'|\mathrm{Y}_{1:k-1})\,\delta\mathrm{X}' \\ &= e^{\mu_b b_h - \mu_b} G_{k-1}[q_s + p_s p_{t,h}], \end{aligned}$$

where $b_h \triangleq \int_{\mathcal{X}} h(\mathrm{x})b(\mathrm{x})d\mathrm{x}$. By *Assumption 4.8*, both the posterior and prior state random finite sets follow multi-object discrete-Gamma i.i.d. cluster processes, i.e., their p.g.fl. take the form of (4.30). Obtaining the first functional derivative of the prior p.g.fl.:

$$\begin{aligned} \frac{\delta}{\delta\mathrm{x}} G_{k|k-1}[h] &= \frac{\delta}{\delta\mathrm{x}} \left(e^{\mu_b b_h - \mu_b} G_{k-1}[q_s + p_s p_{t,h}] \right) \\ &= e^{\mu_b b_h - \mu_b} G_{k-1}[q_s + p_s p_{t,h}] \mu_b \frac{\delta b_h}{\delta\mathrm{x}} \\ &\quad + e^{\mu_b b_h - \mu_b} G^{(1)}_{k-1}[q_s + p_s p_{t,h}] \frac{\delta \langle q_s + p_s p_{t,h}, \varsigma_{k-1}\rangle}{\delta\mathrm{x}} \\ &= e^{\mu_b b_h - \mu_b} G_{k-1}[q_s + p_s p_{t,h}] \mu_b b(\mathrm{x}) \\ &\quad + e^{\mu_b b_h - \mu_b} G^{(1)}_{k-1}[q_s + p_s p_{t,h}] \langle p_s p_t(\mathrm{x}|\cdot), \varsigma_{k-1}\rangle. \end{aligned} \tag{4.67}$$

where $\langle f, \varsigma_{k-1} \rangle = \int_{\mathcal{X}'} f(\mathrm{x}')\varsigma(\mathrm{x}')d\mathrm{x}'$ and $D_{k-1}(\mathrm{x}) = \alpha_{k-1}\beta_{k-1}^{-1}\varsigma_{k-1}(\mathrm{x})$. Set $f(\mathrm{x}') \coloneqq q_s + p_s p_{t,h}$ and recall that

$$G_{k-1}[f] = \frac{\mathrm{Li}_{-\alpha_{k-1}+1}(e^{-\beta_{k-1}}\langle f, \varsigma_{k-1}\rangle)}{\mathrm{Li}_{-\alpha_{k-1}+1}(e^{-\beta_{k-1}})},$$
$$G_{k-1}^{(1)}[f] = \frac{\langle f, \varsigma_{k-1}\rangle^{-1}\mathrm{Li}_{-\alpha_{k-1}}(e^{-\beta_{k-1}}\langle f, \varsigma_{k-1}\rangle)}{\mathrm{Li}_{-\alpha_{k-1}+1}(e^{-\beta_{k-1}})}.$$

For $f = 1$, $\langle 1, \varsigma_{k-1}\rangle = 1$, hence

$$G_{k-1}[1] = \frac{\mathrm{Li}_{-\alpha_{k-1}+1}(e^{-\beta_{k-1}})}{\mathrm{Li}_{-\alpha_{k-1}+1}(e^{-\beta_{k-1}})} = 1,$$
$$G_{k-1}^{(1)}[1] = \frac{\mathrm{Li}_{-\alpha_{k-1}}(e^{-\beta_{k-1}})}{\mathrm{Li}_{-\alpha_{k-1}+1}(e^{-\beta_{k-1}})} \approx \alpha_{k-1}\beta_{k-1}^{-1}.$$

By noticing that $b_{h=1} = \int_{\mathcal{X}} b(\mathrm{x})d\mathrm{x} = 1$ and $p_{t,h=1} = \int_{\mathcal{X}} p_t(\mathrm{x}|\mathrm{x}')d\mathrm{x} = 1$, then the prior intensity function, $D_{k|k-1}(\mathrm{x}) = \frac{\delta}{\delta \mathrm{x}} G_{k|k-1}[1]$, can be obtained from (4.67) as

$$\begin{aligned} D_{k|k-1}(\mathrm{x}) &= e^{\mu_b 1 - \mu_b} G_{k-1}[1]\mu_b b(\mathrm{x}) \\ &\quad + e^{\mu_b 1 - \mu_b} G_{k-1}^{(1)}[1]\langle p_s p_t(\mathrm{x}|\cdot), \varsigma_{k-1}\rangle \\ &\approx 1 \cdot \mu_b b(\mathrm{x}) + 1 \cdot \alpha_{k-1}\beta_{k-1}^{-1}\langle p_s p_t(\mathrm{x}|\cdot), \varsigma_{k-1}\rangle, \\ D_{k|k-1}(\mathrm{x}) &= D_b(\mathrm{x}) + \int_{\mathcal{X}'} p_s(\mathrm{x}')p_t(\mathrm{x}|\mathrm{x}')D_{k-1}(\mathrm{x}')d\mathrm{x}'. \end{aligned}$$

Since the previous posterior and prior state random finite sets are i.i.d. cluster processes, then the probability generating functions of their cardinality distributions are given by $G_{k-1}(z) = G_{k-1}[z]$ and $G_{k|k-1}(z) = G_{k|k-1}[z]$, for $|z| \leq 1$. Take the first two derivatives of $G_{k|k-1}(z)$,

$$\begin{aligned} G_{k|k-1}^{(1)}(z) &= e^{\mu_b z - \mu_b} G_{k-1}[q_s + p_s z]\mu_b \\ &\quad + e^{\mu_b z - \mu_b} G_{k-1}^{(1)}[q_s + p_s z]\langle p_s, \varsigma_{k-1}\rangle, \end{aligned} \tag{4.68}$$

$$\begin{aligned} G_{k|k-1}^{(2)}(z) &= e^{\mu_b z - \mu_b} G_{k-1}[q_s + p_s z]\mu_b^2 \\ &\quad + 2e^{\mu_b z - \mu_b} G_{k-1}^{(1)}[q_s + p_s z]\mu_b\langle p_s, \varsigma_{k-1}\rangle \\ &\quad + e^{\mu_b z - \mu_b} G_{k-1}^{(2)}[q_s + p_s z]\langle p_s, \varsigma_{k-1}\rangle^2, \end{aligned} \tag{4.69}$$

where

$$\begin{aligned} G_{k-1}^{(2)}[q_s + p_s z] &= \frac{\langle q_s + p_s z, \varsigma_{k-1}\rangle^{-2}\mathrm{Li}_{-\alpha_{k-1}-1}(e^{-\beta_{k-1}}\langle q_s + p_s z, \varsigma_{k-1}\rangle)}{\mathrm{Li}_{1-\alpha_{k-1}}(e^{-\beta_{k-1}})} \\ &\quad - \frac{\langle q_s + p_s z, \varsigma_{k-1}\rangle^{-1}\mathrm{Li}_{-\alpha_{k-1}}(e^{-\beta_{k-1}}\langle q_s + p_s z, \varsigma_{k-1}\rangle)}{\mathrm{Li}_{1-\alpha_{k-1}}(e^{-\beta_{k-1}})}. \end{aligned} \tag{4.70}$$

Evaluate (4.68) and (4.69) at $z = 1$, and notice that $G_{k-1}^{(2)}[1] \approx \alpha_{k-1}(\alpha_{k-1}+1)\beta_{k-1}^{-2} - \alpha_{k-1}\beta_{k-1}^{-1}$

to compose the first two cardinality moments:

$$\mu_{N,k|k-1} = G^{(1)}_{k|k-1}(1) \approx \underbrace{\mu_b}_{N_b} + \underbrace{\alpha_{k-1}\beta^{-1}_{k-1}\langle p_s, \varsigma_{k-1}\rangle}_{N_s},$$

$$\begin{aligned}
\sigma^2_{N,k|k-1} &= G^{(2)}_{k|k-1}(1) - \mu^2_{N,k|k-1} + \mu_{N,k|k-1} \\
&\approx \mu_b^2 + 2\mu_b\alpha_{k-1}\beta^{-1}_{k-1}\langle p_s, \varsigma_{k-1}\rangle \\
&+ \left[\alpha_{k-1}(\alpha_{k-1}+1)\beta^{-2}_{k-1} - \alpha_{k-1}\beta^{-1}_{k-1}\right]\langle p_s, \varsigma_{k-1}\rangle^2 \\
&- \left[\mu_b + \alpha_{k-1}\beta^{-1}_{k-1}\langle p_s, \varsigma_{k-1}\rangle\right]^2 + \mu_{N,k|k-1} \\
&= \mu_{N,k|k-1} + \alpha_{k-1}\beta^{-1}_{k-1}\left(\beta^{-1}_{k-1} - 1\right)\langle p_s, \varsigma_{k-1}\rangle^2.
\end{aligned}$$

The prior state RFS is assumed to follow a multi-object discrete-Gamma process with $\mu_{N,k|k-1} \coloneqq \alpha_{k|k-1}\beta^{-1}_{k|k-1}$ and $\sigma^2_{N,k|k-1} \coloneqq \alpha_{k|k-1}\beta^{-2}_{k|k-1}$, for $\alpha_{k|k-1} \in \mathbb{N}$, therefore

$$\alpha_{k|k-1} \approx \frac{\mu^2_{N,k|k-1}}{\sigma^2_{N,k|k-1}},\ \beta_{k|k-1} \approx \frac{\mu_{N,k|k-1}}{\sigma^2_{N,k|k-1}}.$$

□

4.8.2 Proof of Proposition 4.2

Proof of Proposition 4.2. From [129] we invoke the expression for the p.g.fl. of a CPHD posterior process, which is generally valid for *Assumptions 4.4–4.7* and i.i.d. cluster processes, and reads

$$\begin{aligned}
G_k[h] &= \frac{\frac{\delta F[0,h]}{\delta \mathrm{Y}_k}}{\frac{\delta F[0,1]}{\delta \mathrm{Y}_k}} \\
&= \frac{\sum_{j=0}^{m} G_c^{(m-j)}(0)\cdot G^{(j)}_{k|k-1}(\langle hq_d, \varsigma_{k|k-1}\rangle)\sigma_j(\mathrm{Y}_k, h)}{\sum_{i=0}^{m} G_c^{(m-i)}(0)\cdot G^{(i)}_{k|k-1}(\langle\ \ q_d, \varsigma_{k|k-1}\rangle)\sigma_i(\mathrm{Y}_k, 1)},
\end{aligned}$$

where $m = |\mathrm{Y}_k|$, $G_{k|k-1}(\cdot)$ is the p.g.f. of the prior cardinality distribution,

$$\sigma_i(\mathrm{Y}_k, h) = \sigma_{m,i}\left(\frac{\langle hp_d\ell_{\mathrm{y}_1}, \varsigma_{k|k-1}\rangle}{c(\mathrm{y}_1)}, \ldots, \frac{\langle hp_d\ell_{\mathrm{y}_m}, \varsigma_{k|k-1}\rangle}{c(\mathrm{y}_m)}\right)$$

is the elementary symmetric function of degree i in $\frac{\langle hp_d\ell_{\mathrm{y}_1}, \varsigma_{k|k-1}\rangle}{c(\mathrm{y}_1)}, \ldots, \frac{\langle hp_d\ell_{\mathrm{y}_m}, \varsigma_{k|k-1}\rangle}{c(\mathrm{y}_m)}$, and

$$F[g,h] = \int\int h^{\mathrm{X}} g^{\mathrm{Y}} p_\Psi(\mathrm{Y}_k|\mathrm{X})\, p_{\Xi,k|k-1}(\mathrm{X}|\mathrm{Y}_{1:k-1})\,\delta\mathrm{X}\delta\mathrm{Y}$$

is the joint p.g.fl. on the state and observation random finite sets. First, we invoke another result from [129], the posterior intensity function estimated by the CPHD filter according to:

$$D_k(\mathrm{x}) \approx \frac{q_d(\mathrm{x})}{G^{(1)}_{k|k-1}(1)}\Upsilon_k[\mathrm{Y}_k]\, D_{k|k-1}(\mathrm{x}) + \frac{p_d(\mathrm{x})}{G^{(1)}_{k|k-1}(1)}\sum_{\mathrm{y}\in\mathrm{Y}_k}\frac{\ell_{\mathrm{y}}(\mathrm{x})}{c(\mathrm{y})}\Upsilon_k[\mathrm{Y}_k \setminus \{\mathrm{y}\}]\, D_{k|k-1}(\mathrm{x}),$$

with

$$\Upsilon_k[\mathrm{Y}] = \frac{\sum_{j=0}^{|\mathrm{Y}|} G_c^{(|\mathrm{Y}|-j)}(0) G_{k|k-1}^{(j+1)}(\langle q_d, \varsigma_{k|k-1}\rangle)\sigma_j(\mathrm{Y}\ ,1)}{\sum_{i=0}^{m}\ G_c^{(m-i)}(0) G_{k|k-1}^{(i)}\left(\langle q_d, \varsigma_{k|k-1}\rangle\right)\sigma_i(\mathrm{Y}_k,1)}, \tag{4.71}$$

whose proof follows from evaluating $D_k(\mathrm{x}) = \frac{\delta}{\delta \mathrm{x}} G_k[1]$ (see [129]). In the DG-CPHD filter context, we use *Assumptions 4.5* (clutter RFS follows a Poisson point process) and *4.8* to further simplify (4.71). The p.g.fl. of the clutter process is given by $G_c[g] = \exp\left(\lambda \int_{\mathcal{Z}} g(\mathrm{y}) c(\mathrm{y}) d\mathrm{y} - \lambda\right)$. Set $G_c^{(i)}(z) \coloneqq e^{\lambda z - \lambda}$ and notice that $G_c^{(\ell)}(0) = \lambda^{\ell} e^{-\lambda}$ to obtain

$$\begin{aligned}
\Upsilon_k[\mathrm{Y}] &= \frac{\lambda^{|\mathrm{Y}|}\sum_{j=0}^{|\mathrm{Y}|}\lambda^{-j} G_{k|k-1}^{(j+1)}(\langle q_d, \varsigma_{k|k-1}\rangle)\sigma_j(\mathrm{Y}\ ,1)}{\lambda^{m}\sum_{i=0}^{m}\ \lambda^{-i} G_{k|k-1}^{(i)}\left(\langle q_d, \varsigma_{k|k-1}\rangle\right)\sigma_i(\mathrm{Y}_k,1)} \\
&= \frac{\lambda^{|\mathrm{Y}|}\sum_{j=0}^{|\mathrm{Y}|} G_{k|k-1}^{(j+1)}(\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_j(\mathrm{Y}\ )}{\lambda^{m}\sum_{i=0}^{m}\ G_{k|k-1}^{(i)}\left(\langle q_d, \varsigma_{k|k-1}\rangle\right)\bar{\sigma}_i(\mathrm{Y}_k)} \\
&\triangleq \lambda^{|\mathrm{Y}|-m}\Theta_k[\mathrm{Y}],
\end{aligned} \tag{4.72}$$

where

$$\bar{\sigma}_i(\mathrm{Y}) \triangleq \sigma_{m=|\mathrm{Y}|,i}\left(\frac{\langle p_d \ell_{\mathrm{y}_1}, \varsigma\rangle}{\lambda c(\mathrm{y}_1)}, \ldots, \frac{\langle p_d \ell_{\mathrm{y}_m}, \varsigma\rangle}{\lambda c(\mathrm{y}_m)}\right). \tag{4.73}$$

From (4.72), (4.73), and observing that $\Upsilon_k[\mathrm{Y}_k] = \Theta_k[\mathrm{Y}_k]$ and $\Upsilon_k[\mathrm{Y}_k \setminus \{\mathrm{y}\}] = \lambda^{-1}\Theta_k[\mathrm{Y}_k \setminus \{\mathrm{y}\}]$, the posterior intensity for the DG-CPHD recursion, according to (4.35), follows straightforwardly.

As follows, we use *Assumption 4.8* to derive the posterior cardinality parameters. Under this assumption, the posterior state RFS follows an i.i.d. cluster process and so the posterior cardinality p.g.f. is given by $G_k(z) = G_k[z]$, which results

$$\begin{aligned}
G_k(z) &= \frac{\sum_{j=0}^{m}\lambda^{-j} G_{k|k-1}^{(j)}(z\langle q_d, \varsigma_{k|k-1}\rangle)\sigma_j(\mathrm{Y}_k, z)}{\sum_{i=0}^{m}\lambda^{-i} G_{k|k-1}^{(i)}(\ \langle q_d, \varsigma_{k|k-1}\rangle)\sigma_i(\mathrm{Y}_k,1)} \\
&= \frac{\sum_{j=0}^{m} z^j G_{k|k-1}^{(j)}(z\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_i(\mathrm{Y}_k)}{\sum_{i=0}^{m}\quad G_{k|k-1}^{(i)}(\ \langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_j(\mathrm{Y}_k)}.
\end{aligned} \tag{4.74}$$

We compute the first two derivatives of $G_k(z)$ as

$$\begin{aligned}
G_k^{(1)}(z) = &\frac{\sum_{j=0}^{m} j\cdot z^{j-1} G_{k|k-1}^{(j)}(z\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_j(\mathrm{Y}_k)}{\sum_{i=0}^{m} G_{k|k-1}^{(i)}(\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_i(\mathrm{Y}_k)} \\
&+ \frac{\sum_{j=0}^{m} z^j G_{k|k-1}^{(j+1)}(z\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_j(\mathrm{Y}_k)}{\sum_{i=0}^{m} G_{k|k-1}^{(i)}(\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_i(\mathrm{Y}_k)}\langle q_d, \varsigma_{k|k-1}\rangle,
\end{aligned} \tag{4.75}$$

$$G_k^{(2)}(z) = \frac{\sum_{j=0}^{m} j(j-1)\cdot z^{j-2} G_{k|k-1}^{(j)}(z\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_j(\mathrm{Y}_k)}{\sum_{i=0}^{m} G_{k|k-1}^{(i)}(\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_i(\mathrm{Y}_k)} + 2\frac{\sum_{j=0}^{m} j\cdot z^{j-1} G_{k|k-1}^{(j+1)}(z\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_j(\mathrm{Y}_k)}{\sum_{i=0}^{m} G_{k|k-1}^{(i)}(\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_i(\mathrm{Y}_k)}\langle q_d, \varsigma_{k|k-1}\rangle + \frac{\sum_{j=0}^{m} z^{j} G_{k|k-1}^{(j+2)}(z\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_j(\mathrm{Y}_k)}{\sum_{i=0}^{m} G_{k|k-1}^{(i)}(\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_i(\mathrm{Y}_k)}\langle q_d, \varsigma_{k|k-1}\rangle^2. \tag{4.76}$$

Evaluate (4.75) and (4.76) at $z = 1$ to obtain

$$G_k^{(1)}(1) = \frac{\sum_{j=0}^{m} j\cdot G_{k|k-1}^{(j)}(\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_j(\mathrm{Y}_k)}{\sum_{i=0}^{m} G_{k|k-1}^{(i)}(\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_i(\mathrm{Y}_k)} + \frac{\sum_{j=0}^{m} G_{k|k-1}^{(j+1)}(\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_j(\mathrm{Y}_k)}{\sum_{i=0}^{m} G_{k|k-1}^{(i)}(\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_i(\mathrm{Y}_k)}\langle q_d, \varsigma_{k|k-1}\rangle,$$

$$G_k^{(2)}(1) = \frac{\sum_{j=0}^{m} j^2 G_{k|k-1}^{(j)}(\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_j(\mathrm{Y}_k)}{\sum_{i=0}^{m} G_{k|k-1}^{(i)}(\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_i(\mathrm{Y}_k)} - \frac{\sum_{j=0}^{m} j\, G_{k|k-1}^{(j)}(\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_j(\mathrm{Y}_k)}{\sum_{i=0}^{m} G_{k|k-1}^{(i)}(\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_i(\mathrm{Y}_k)} + 2\frac{\sum_{j=0}^{m} j\, G_{k|k-1}^{(j+1)}(\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_j(\mathrm{Y}_k)}{\sum_{i=0}^{m} G_{k|k-1}^{(i)}(\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_i(\mathrm{Y}_k)}\langle q_d, \varsigma_{k|k-1}\rangle + \frac{\sum_{j=0}^{m} G_{k|k-1}^{(j+2)}(y\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_j(\mathrm{Y}_k)}{\sum_{i=0}^{m} G_{k|k-1}^{(i)}(\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_i(\mathrm{Y}_k)}\langle q_d, \varsigma_{k|k-1}\rangle^2.$$

By defining

$$\theta_{u,v} \triangleq \frac{\sum_{j=0}^{m_k} j^u G_{k|k-1}^{(j+v)}(\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_j(\mathrm{Y}_k)}{\sum_{i=0}^{m_k} \quad G_{k|k-1}^{(i)}(\langle q_d, \varsigma_{k|k-1}\rangle)\bar{\sigma}_i(\mathrm{Y}_k)},$$

one has

$$G_k^{(1)}(1) = \theta_{1,0} + \theta_{0,1}\cdot\langle q_d, \varsigma_{k|k-1}\rangle,$$
$$G_k^{(2)}(1) = \theta_{2,0} - \theta_{1,0} + 2\theta_{1,1}\cdot\langle q_d, \varsigma_{k|k-1}\rangle + \theta_{0,2}\cdot\langle q_d, \varsigma_{k|k-1}\rangle^2,$$

from which the posterior cardinality moments are obtained as given in (4.37) and (4.38). The posterior state RFS is assumed to follow a multi-object discrete-Gamma process with cardinality characterized by $\mu_{N,k} \coloneqq \alpha_k\beta_k^{-1}$ and $\sigma_{N,k}^2 \coloneqq \alpha_k\beta_k^{-2}$, for $\alpha_k \in \mathbb{N}$, therefore (4.36) holds. Since

the prior cardinality distribution is also assumed to be a discrete-Gamma distribution, we write

$$\begin{aligned} G_{k|k-1}^{(\ell)}(z) &= \frac{d^\ell}{dz^\ell}\left(\frac{\mathrm{Li}_{-\alpha_{k|k-1}+1}(e^{-\beta_{k|k-1}}z)}{\mathrm{Li}_{-\alpha_{k|k-1}+1}(e^{-\beta_{k|k-1}}\ )}\right) \\ G_{k|k-1}^{(\ell)}(z) &= \frac{\frac{d^\ell}{dz^\ell}\left(\mathrm{Li}_{-\alpha_{k|k-1}+1}(e^{-\beta_{k|k-1}}z)\right)}{\mathrm{Li}_{-\alpha_{k|k-1}+1}(e^{-\beta_{k|k-1}})}, \\ &\triangleq \frac{\hat{G}_{k|k-1}^{(\ell)}(z)}{\hat{G}_{k|k-1}^{(0)}(1)}. \end{aligned}$$

The proof is complete by noting that, in all terms of (4.75) and (4.76), $1/\hat{G}_{k|k-1}^{(0)}(1)$ appears both in the numerator and denominator and can be cancelled out to leave only terms depending on $\hat{G}_{k|k-1}^{(\ell)}(z)$. □

4.8.3 Proof of Proposition 4.3

Lemma 4.4. *Let $f : [0,\infty) \to [0,\infty)$ be a nonnegative increasing function, such that $0 < \sum_{\ell=0}^{L} f(\ell) < \infty$ for $L \in \mathbb{Z}_+$. If there is a nonnegative function $g : [0,\infty) \to [0,\infty)$ satisfying $g(\ell) = \mathcal{O}(f(\ell))$ such that $\frac{f(\ell)}{\sum_{\ell=0}^{L} f(\ell)} \geq \frac{g(\ell)}{\sum_{\ell=0}^{L} g(\ell)}$ for $\ell \geq L_0$, $L_0 \in \mathbb{R}_+$, then*

$$\frac{\sum_{\ell=0}^{L} \ell \cdot f(\ell)}{\sum_{\ell=0}^{L} f(\ell)} \geq \frac{\sum_{\ell=0}^{L} \ell \cdot g(\ell)}{\sum_{\ell=0}^{L} g(\ell)}, \tag{4.77}$$

for $L \geq L_0 + \frac{L_0 - \lfloor L_0 \rfloor}{2}$.

Proof. Write $w_{f,\ell} \coloneqq f(\ell)/\sum_{\ell=0}^{L} f(\ell)$ and $w_{g,\ell} \coloneqq g(\ell)/\sum_{\ell=0}^{L} g(\ell)$. Note that $\sum_{\ell=0}^{L} w_{f,\ell} = \sum_{\ell=0}^{L} w_{g,\ell} = 1$ and so

$$\begin{aligned} 0 \leq \sum_{\ell=\lceil L_0 \rceil}^{L} (w_{f,\ell} - w_{g,\ell}) &= \sum_{\ell=0}^{\lfloor L_0 \rfloor} (w_{g,\ell} - w_{f,\ell}) < 1, \\ \sum_{\ell=\lceil L_0 \rceil}^{L} L_0 (w_{f,\ell} - w_{g,\ell}) &= \sum_{\ell=0}^{\lfloor L_0 \rfloor} L_0 (w_{g,\ell} - w_{f,\ell}), \\ \sum_{\ell=\lceil L_0 \rceil}^{L} \ell (w_{f,\ell} - w_{g,\ell}) &\geq \sum_{\ell=0}^{\lfloor L_0 \rfloor} \ell (w_{g,\ell} - w_{f,\ell}). \end{aligned} \tag{4.78}$$

Therefore $\sum_{\ell=0}^{L} \ell \cdot w_{f,\ell} \geq \sum_{\ell=0}^{L} \ell \cdot w_{g,\ell}$, which corresponds to (4.77). The condition $L \geq L_0 + (L_0 - \lfloor L_0 \rfloor)/2$ must be met to ensure $2(L - L_0)M \geq (L_0 - \lfloor L_0 \rfloor)M$, where

$$M = \frac{(w_{f,L} - w_{g,L}) - (w_{f,\lfloor L_0 \rfloor} - w_{g,\lfloor L_0 \rfloor})}{L - \lfloor L_0 \rfloor},$$

so that $2(w_{f,2} - w_{g,2}) \geq 1(w_{g,1} - w_{f,1})$ in the particular case when $\lfloor L_0 \rfloor = 1$ and $L = 2$. □

Proof of Proposition 4.3. We resort to a special type of Chebyshev's inequality that is appro-

priate for bounding the probabilities of non-symmetric intervals, [72], viz.

$$\varepsilon = \Pr\{0 < n < n_{\max}\} \geq 4\frac{\mu_{N,k} n_{\max} - (\mu_{N,k}^2 + \sigma_{N,k}^2)}{n_{\max}^2}. \tag{4.79}$$

where $0 < \sigma_{N,k}^2 \leq \mu_{N,k} n_{\max} - \mu_{N,k}^2$. Rearranging terms and solving the inequality for $n_{\max}$ corresponding to the upper root:

$$n_{\max} \geq \frac{2\mu_{N,k}}{\varepsilon}\left(1 + \sqrt{(1-\varepsilon) - \varepsilon\left(\frac{\sigma_{N,k}}{\mu_{N,k}}\right)^2}\right). \tag{4.80}$$

As follows, we bound $\mu_{N,k}$ as per the moment update equation [203]:

$$\begin{aligned}
\mu_{N,k} &= \frac{\sum_{n\geq 0} n \cdot \sum_{j=0}^{\min(m,n)} G_c^{(m-j)}(0) n^{\underline{j}} q_d^{n-j} \sigma_j(\mathrm{Y}) \cdot p_{k|k-1}(n)}{\sum_{n\geq 0} \sum_{j=0}^{\min(m,n)} G_c^{(m-j)}(0) n^{\underline{j}} q_d^{n-j} \sigma_j(\mathrm{Y}) \cdot p_{k|k-1}(n)} \\
&= \frac{\sum_{n\geq 0} n \cdot \sum_{j=0}^{m} \mathbb{1}_{m\leq n} \lambda^{m-j} e^{-\lambda} n^{\underline{j}} q_d^{n-j} \sigma_j(\mathrm{Y}) \cdot p_{k|k-1}(n)}{\sum_{n\geq 0} \sum_{j=0}^{m} \mathbb{1}_{m\leq n} \lambda^{m-j} e^{-\lambda} n^{\underline{j}} q_d^{n-j} \sigma_j(\mathrm{Y}) \cdot p_{k|k-1}(n)} \\
&\geq \frac{\sum_{j=0}^{m} j \cdot \mathbb{E}_{N,k|k-1}\left[\mathbb{1}_{m\leq n} \lambda^{-j} n^{\underline{j}} q_d^{n-j}\right] \sigma_j(\mathrm{Y})}{\sum_{j=0}^{m} \quad \mathbb{E}_{N,k|k-1}\left[\mathbb{1}_{m\leq n} \lambda^{-j} n^{\underline{j}} q_d^{n-j}\right] \sigma_j(\mathrm{Y})} \\
&\geq \frac{\sum_{j=0}^{m} j \cdot \mathbb{1}_{m\leq n^\star} \lambda^{-j} \mu_{N,k|k-1}^{\underline{j}} q_d^{\mu_{N,k|k-1}-j} \sigma_j(\mathrm{Y})}{\sum_{j=0}^{m} \quad \mathbb{1}_{m\leq n^\star} \lambda^{-j} \mu_{N,k|k-1}^{\underline{j}} q_d^{\mu_{N,k|k-1}-j} \sigma_j(\mathrm{Y})},
\end{aligned} \tag{4.81}$$

where $n^{\underline{j}} \triangleq n(n-1)\ldots(n-j+1)$ is the Pochhammer's symbol and

$$n^\star \triangleq \min\left(\mu_{N,k|k-1}, \arg\max_j \sigma_j(\mathrm{Y})\right).$$

Per Jensen's inequality

$$\begin{aligned}
\mathbb{E}_{N,k|k-1}\left[\mathbb{1}_{m\leq n}\varphi_j(n)\right] &\geq \mathbb{E}_{N,k|k-1}\left[\mathbb{1}_{m\leq n^\star}\varphi_j(n)\right] \\
&\geq \mathbb{1}_{m\leq n^\star}\varphi_j(\mu_{N,k|k-1}),
\end{aligned}$$

where $\varphi_j(n) \coloneqq \lambda^{-j} n^{\underline{j}} q_d^{n-j}$ is convex in n, and *Lemma 4.4* is applied in (4.81) with $f(j) = \mathbb{E}_{N,k|k-1}\left[\mathbb{1}_{m\leq n^\star}\varphi_j(n)\right]\sigma_j(\mathrm{Y})$ and $g(j) = \mathbb{1}_{m\leq n^\star}\varphi_j(\mu_{N,k|k-1})\sigma_j(\mathrm{Y})$ for $j \in [0, n^\star]$. The bound given by (4.81) is tight and constitutes a close approximation of $\mu_{N,k}$. This bound can be simplified by applying *Lemma 4.4* once more as $\varphi_j(\mu_{N,k|k-1})\sigma_j(\mathrm{Y})$ is a nonnegative increasing function in the interval $j \in [0, n^\star]$ to give

$$\begin{aligned}
\frac{\sum_{j=0}^{n^\star} j \cdot \varphi_j(\mu_{N,k|k-1})\sigma_j(\mathrm{Y})}{\sum_{j=0}^{n^\star} \quad \varphi_j(\mu_{N,k|k-1})\sigma_j(\mathrm{Y})} &\gtrapprox \frac{\sum_{j=0}^{\mu_{N,k|k-1}} j \cdot \inf_j\left[\varphi_j(\mu_{N,k|k-1})\sigma_j(\mathrm{Y})\right]}{\sum_{j=0}^{\mu_{N,k|k-1}} \quad \inf_j\left[\varphi_j(\mu_{N,k|k-1})\sigma_j(\mathrm{Y})\right]} \\
&= \frac{\sum_{j=0}^{\mu_{N,k|k-1}} j}{\sum_{j=0}^{\mu_{N,k|k-1}} 1} = \frac{\mu_{N,k|k-1}}{2}.
\end{aligned} \tag{4.82}$$

Substituting this bound in (4.80) we conclude that there exists a $K \geq 1$ such that $n_{\max} \geq K \cdot \mu_{N,k|k-1}$. □

Remark 4.5. Note that for a certain $x \in \mathbb{R}_+$ such that

$$\prod_{j=1}^{m}\left(1-\frac{\langle p_d \ell_{\mathrm{y}_j}, \varsigma_{k|k-1}\rangle}{c(\mathrm{y}_j)}\right) = (1-x)^m = \sum_{j=0}^{m}\binom{m}{j}x^j$$

$j_{\max} \approx \arg\max_j \sigma_j(\mathrm{Y})$ can be computed as

$$\begin{aligned}
\frac{d}{dj}\left[\binom{m}{j}x^j\right] = \left[\frac{d}{dj}\binom{m}{j}\right]x^j + \binom{m}{j}x^j \log x = 0,\\
-\binom{m}{j}\left(\sum_{\ell=1}^{j}\ell^{-1} - \sum_{\ell=1}^{m-j}\ell^{-1}\right) + \binom{m}{j}\log x = 0,\\
\sum_{\ell=2}^{j}\ell^{-1} - \sum_{\ell=2}^{m-j}\ell^{-1} = \log x,\\
\log j - \log(m-j) \approx \log x,\\
j_{\max} \approx m\frac{x}{1+x},
\end{aligned}$$

where

$$\begin{aligned}
\frac{d}{dj}\binom{m}{j} &= m!\frac{-D_j(j!)\cdot(m-j)! - (j!)D_j((m-j)!)}{[j!(m-j)!]^2}\\
&= -\binom{m}{j}\frac{j!\left(-\gamma+\sum_{\ell=1}^{j}\ell^{-1}\right)(m-j)!}{j!(m-j)!} - \binom{m}{j}\frac{-j!(m-j)!\left(-\gamma+\sum_{\ell=1}^{m-j}\ell^{-1}\right)}{j!(m-j)!}\\
&= -\binom{m}{j}\left(\sum_{\ell=1}^{j}\ell^{-1} - \sum_{\ell=1}^{m-j}\ell^{-1}\right),
\end{aligned}$$

and γ is the Euler–Mascheroni constant.

References

[3] T. M. Apostol. "An Elementary View of Euler's Summation Formula". In: *The American Mathematical Monthly* 106.5 (1999), pp. 409–418.

[18] D. S. Bryant et al. "The CPHD Filter With Target Spawning". In: *IEEE Transactions on Signal Processing* 65.5 (Mar. 2017), pp. 13124–13138.

[27] S. Chakraborty and D. Chakravarty. "Discrete Gamma Distributions: Properties and Parameter Estimations". In: *Communications in Statistics - Theory and Methods* 41.18 (2012), pp. 3301–3324.

[32] P. C. Consul and G. C. Jain. "A Generalization of the Poisson Distribution". In: *Technometrics* 15.4 (1973), pp. 791–799.

[33] P. Consul, S. Kotz, and F. Famoye. *Lagrangian Probability Distributions*. Birkhäuser Boston, 2006.

[44] D. J. Daley and D. Vere-Jones. *An Introduction to the Theory of Point Processes*. Vol. Volume I: Elementary Theory and Methods. Probability and its Applications. New York, Berlin, Paris: Springer, 2003.

[45] D. J. Daley and D. Vere-Jones. *An Introduction to the Theory of Point Processes*. Vol. Volume II: General Theory and Structure. Probability and its Applications. New York, Berlin, Paris: Springer, 2008.

[57] P. Del Moral and J. Houssineau. "Particle Association Measures and Multiple Target Tracking". In: *Theoretical Aspects of Spatial-Temporal Modeling*. Ed. by G. W. Peters and T. Matsui. Tokyo: Springer Japan, 2015, pp. 1–30.

[59] E. Delande et al. "Regional Variance for Multi-Object Filtering". In: *IEEE Transactions on Signal Processing* 62.13 (July 2014), pp. 3415–3428.

[68] O. Erdinc, P. Willett, and Y. Bar-Shalom. "Probability Hypothesis Density Filter for Multitarget Multisensor Tracking". In: *FUSION 2005, Proceedings of the 8th International Conference on Information Fusion*. Vol. 1. July 2005, p. 8.

[72] K. Ferentinos. "On Tchebycheff's type inequalities". In: *Trabajos de Estadística y de Investigación Operativa* 33.1 (1982), pp. 125–132.

[73] T. E. Fortmann, Y. Bar-Shalom, and M. Scheffe. "Multi-target tracking using joint probabilistic data association". In: *19th IEEE Conference on Decision and Control including the Symposium on Adaptive Processes*. Dec. 1980, pp. 807–812.

[125] R. P. S. Mahler. *Advances in Statistical Multisource-Multitarget Information Fusion*. Needham, MA, USA: Artech House, Inc., 2014.

[126] R. P. S. Mahler. "Multitarget Bayes Filtering via First-Order Multitarget Moments". In: *IEEE Transactions on Aerospace and Electronic Systems* 39.4 (Oct. 2003), pp. 1152–1178.

[127] R. P. S. Mahler. "A theory of PHD filters of higher order in target number". In: *Proc. SPIE 6235, Signal Processing, Sensor Fusion, and Target Recognition XV*. Vol. 6235. May 2006, 62350K–62350K-12.

[128] R. P. S. Mahler. *An Introduction to Multisource-Multitarget Statistics and Its Applications*. Technical Monograph. Eagan MN: Lockheed Martin, 2000.

[129] R. P. S. Mahler. "PHD Filters of Higher Order in Target Number". In: *IEEE Transactions on Aerospace and Electronic Systems* 43.4 (Oct. 2007), pp. 1523–1543.

[130] R. P. S. Mahler. "PHD Filters of Second Order in Target Number". In: *Proc. SPIE 6236, Signal and Data Processing of Small Targets*. Vol. 6236. May 2006, 62360P–62360P-12.

[131] R. P. S. Mahler. *Statistical Multisource-Multitarget Information Fusion*. Norwood, MA, USA: Artech House, Inc., 2007.

[143] J. E. Moyal. "The General Theory of Stochastic Population Processes". In: *Acta Mathematica* 108 (1962), pp. 1–31.

[163] D. B. Reid. "An Algorithm for Tracking Multiple Targets". In: *IEEE Transactions on Automatic Control* 24.6 (Dec. 1979), pp. 843–854.

[173] I. Schlangen et al. "A Second-Order PHD Filter With Mean and Variance in Target Number". In: *IEEE Transactions on Signal Processing* 66.1 (Jan. 2018), pp. 48–63.

[174] D. Schuhmacher, B.-T. Vo, and B.-N. Vo. "A Consistent Metric for Performance Evaluation of Multi-Object Filters". In: *IEEE Transactions on Signal Processing* 56.8 (Aug. 2008), pp. 3447–3457.

[176] G. Shmueli et al. "A useful distribution for fitting discrete data: revival of the Conway-Maxwell-Poisson distribution". In: *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 54.1 (2005), pp. 127–142.

[201] B.-N. Vo and W. K. Ma. "The Gaussian Mixture Probability Hypothesis Density Filter". In: *IEEE Transactions on Signal Processing* 54.11 (Nov. 2006), pp. 4091–4104.

[203] B.-T. Vo, B.-N. Vo, and A. Cantoni. "Analytic Implementations of the Cardinalized Probability Hypothesis Density Filter". In: *IEEE Transactions on Signal Processing* 55.7 (July 2007), pp. 3553–3567.

[205] H. W. Watson and F. Galton. "On the Probability of the Extinction of Families". In: *The Journal of the Anthropological Institute of Great Britain and Ireland* 4 (1875), pp. 138–144.

[209] D. Wood. *The Computation of Polylogarithms*. Tech. rep. 15-92*. University of Kent, Canterbury, UK: University of Kent, Computing Laboratory, June 1992, pp. 182–196.

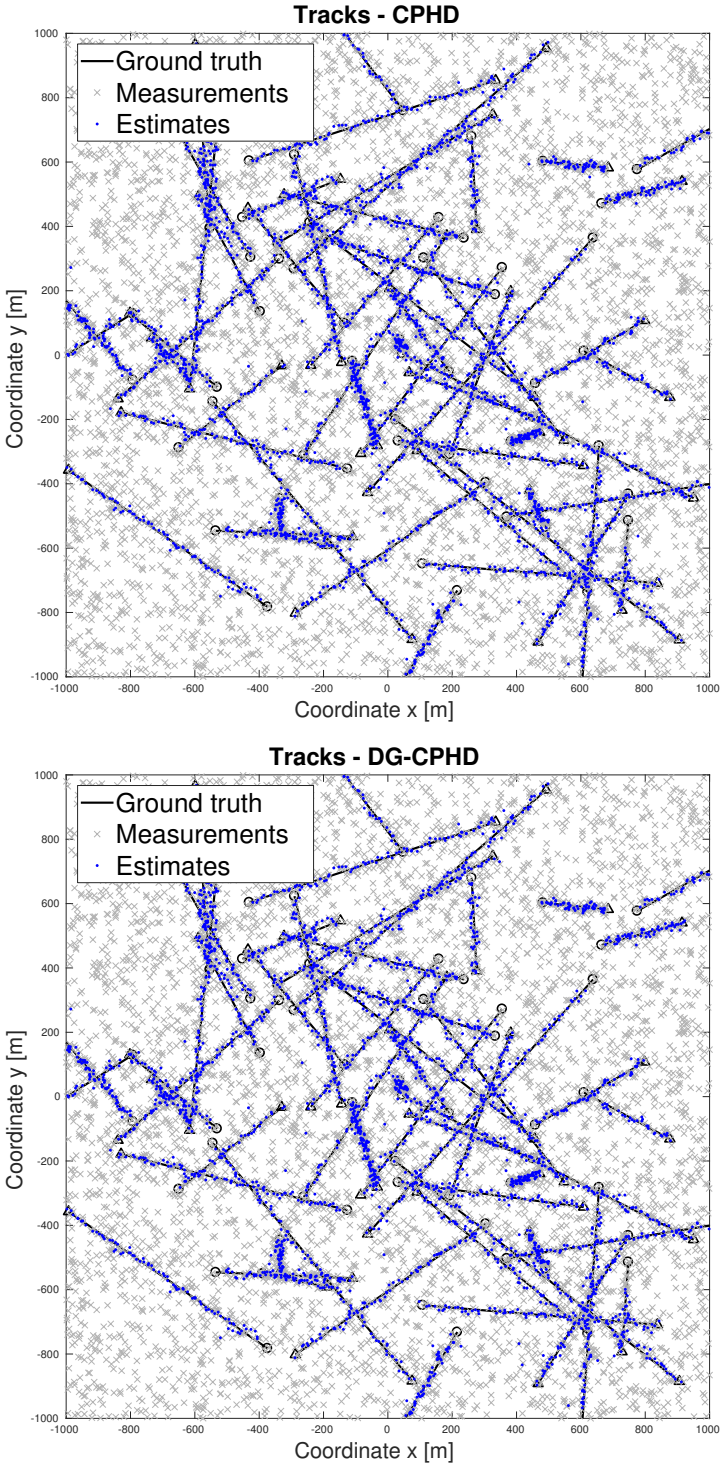


Figure 4.2: Exemplar run: tracks estimated by CPHD and DG-CPHD filters

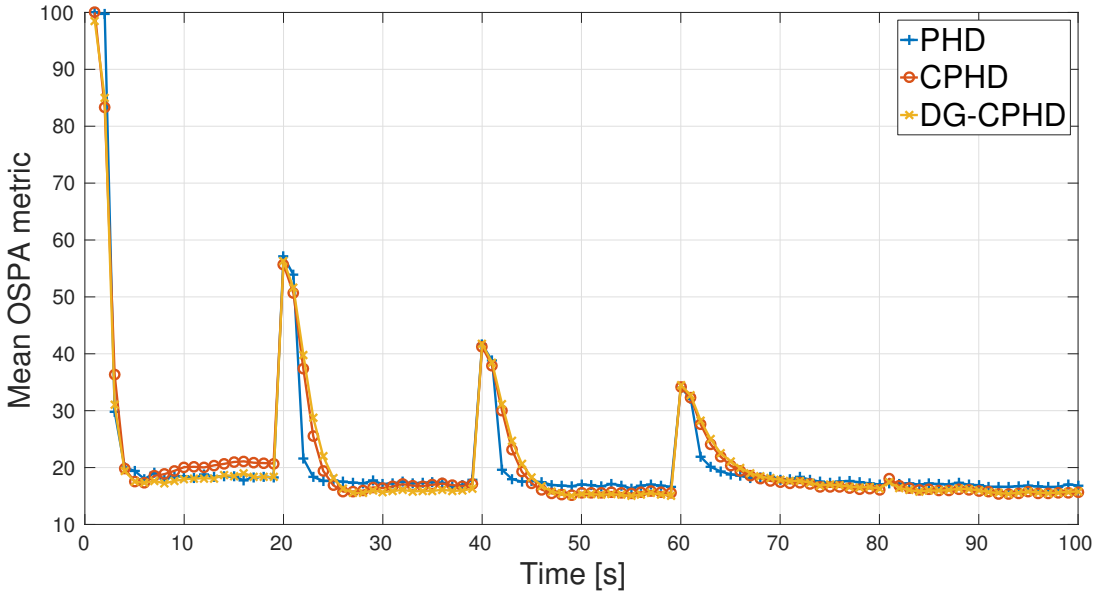


Figure 4.3: MOSPA metric over time ($p_d = 0.98$, $\lambda = 50$, $N_t = 50$)

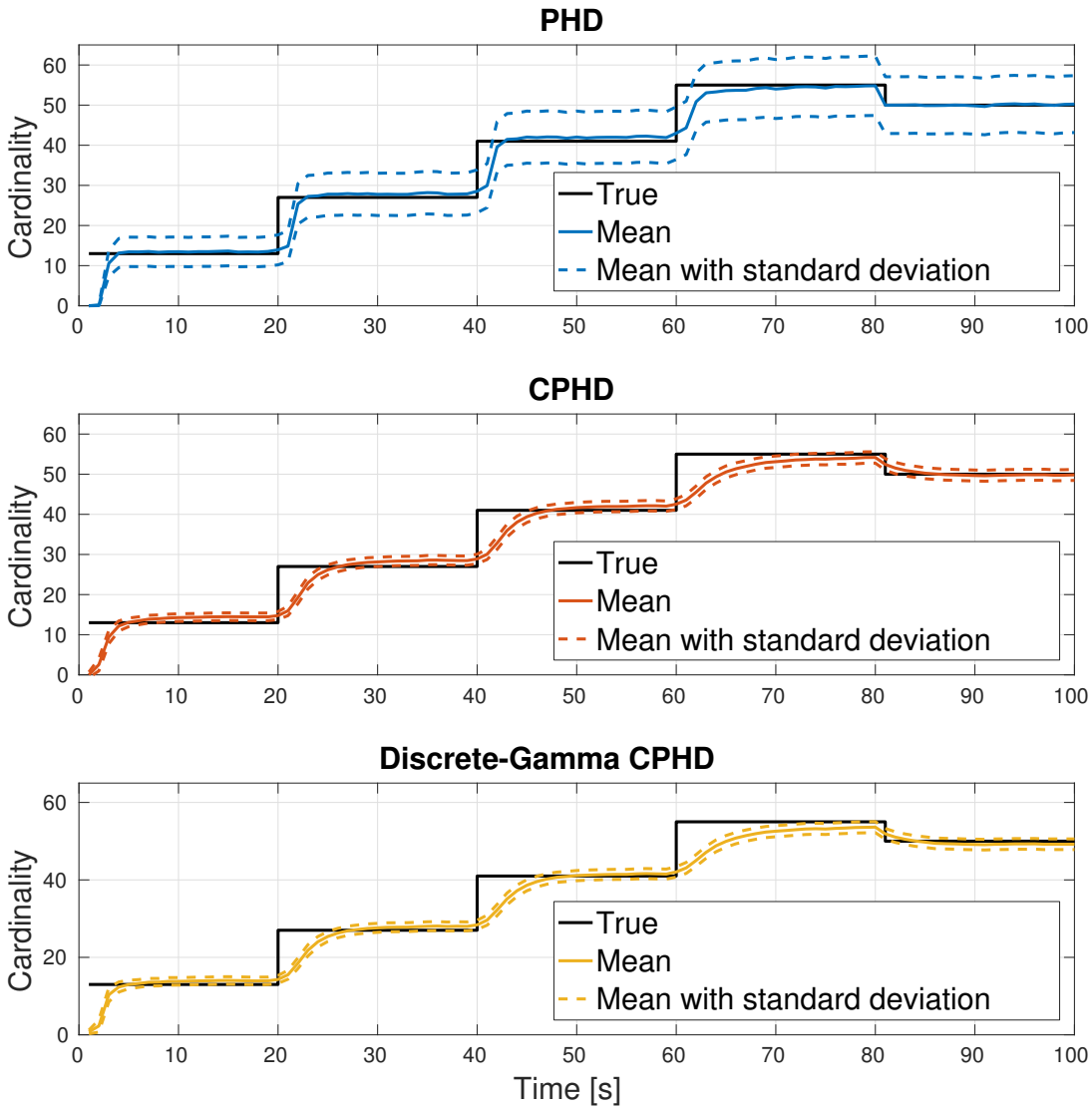


Figure 4.4: Cardinality statistics over time ($p_d = 0.98$, $\lambda = 50$, $N_t = 50$)

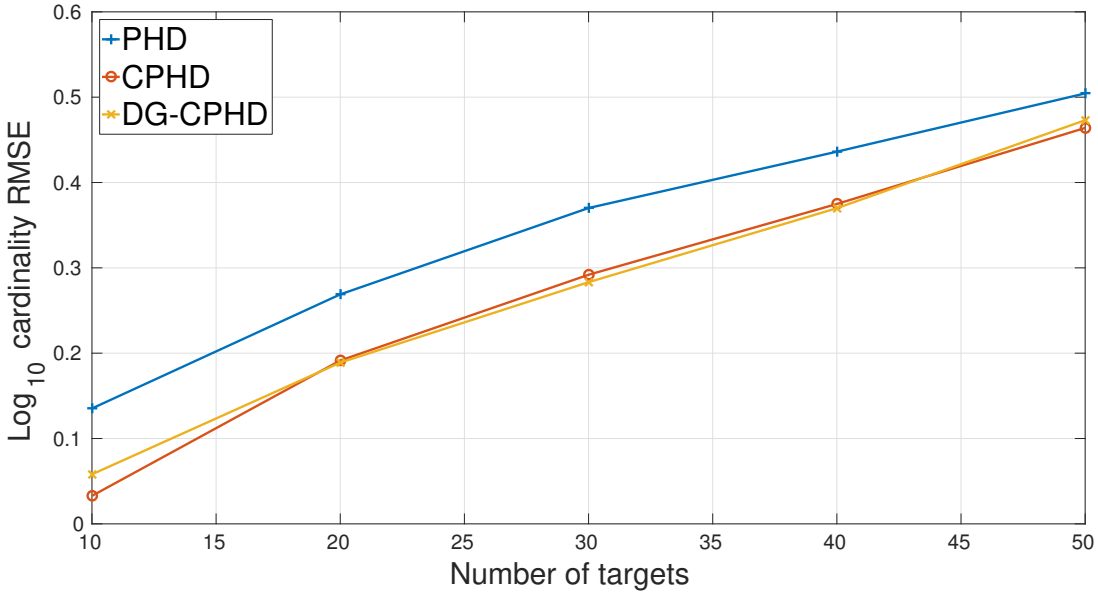


Figure 4.5: Average cardinality RMSE versus number of targets

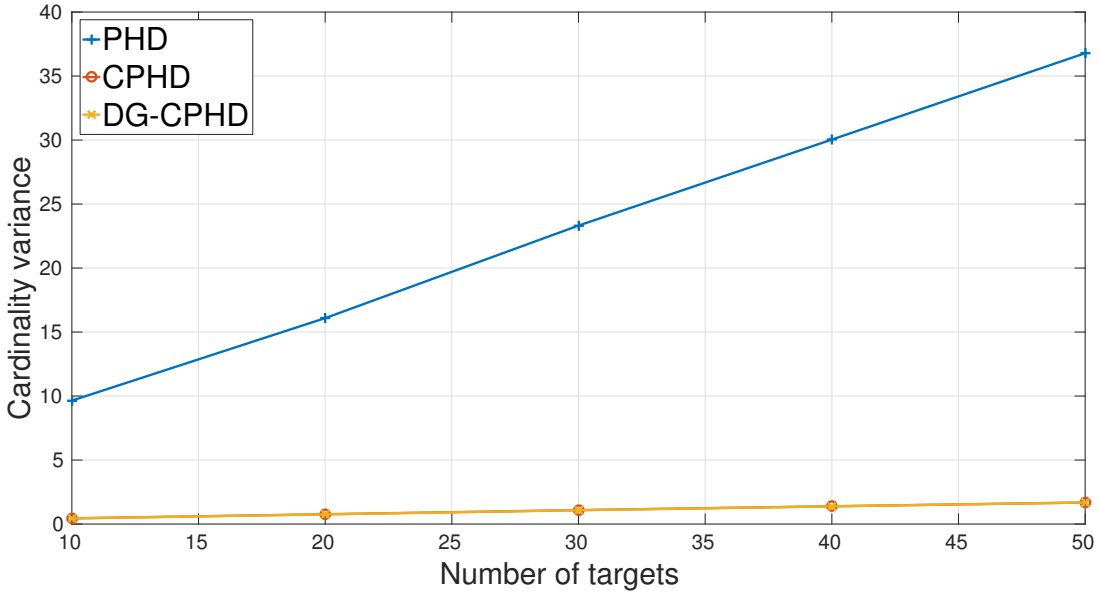


Figure 4.6: Average cardinality variance versus number of targets

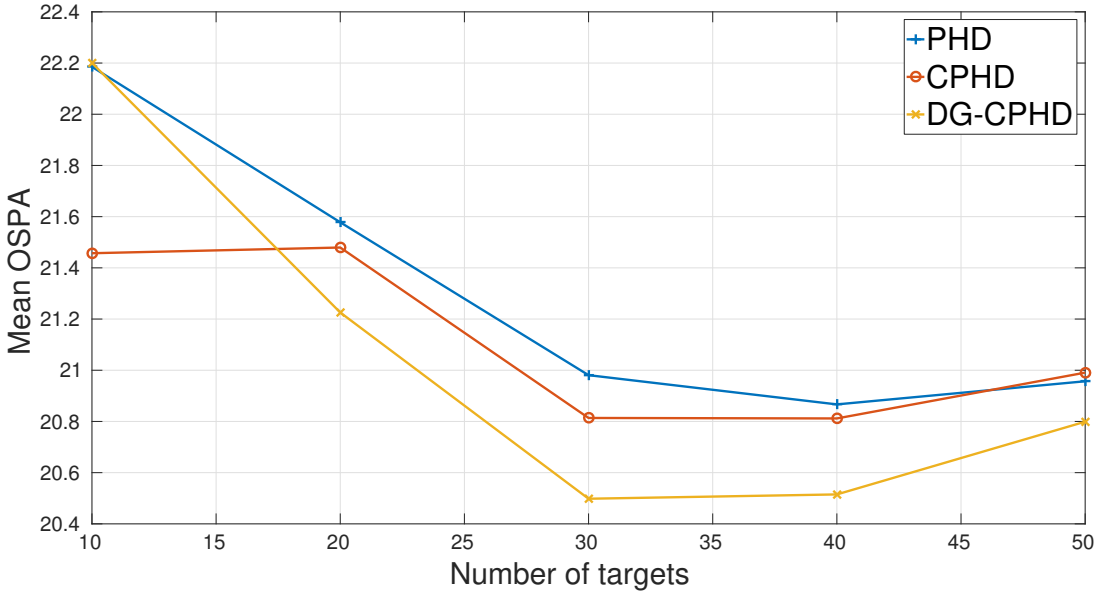


Figure 4.7: Average mean OSPA versus number of targets

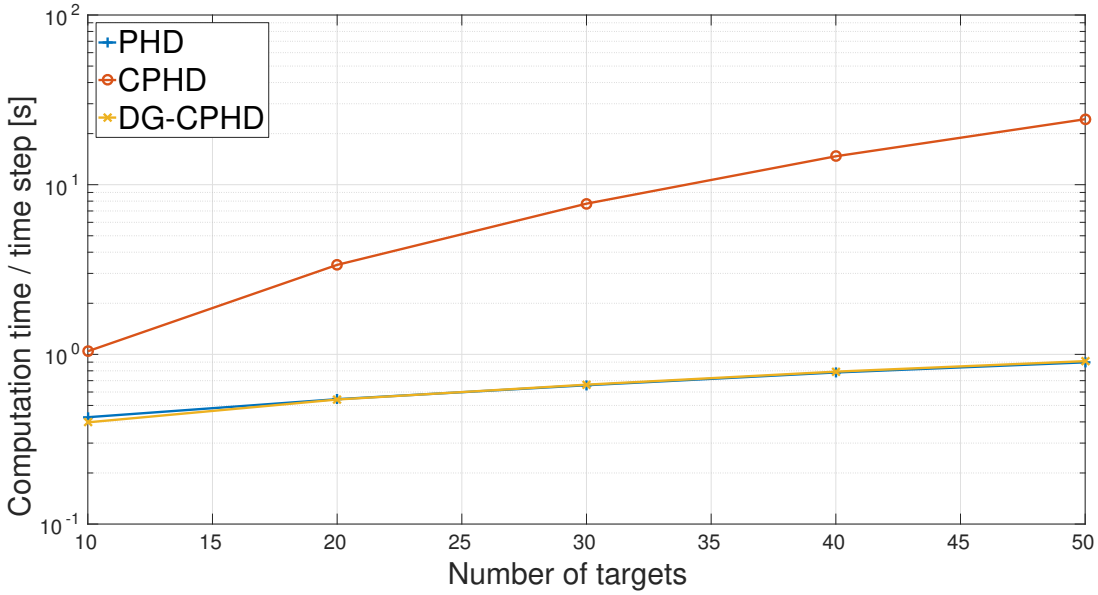


Figure 4.8: Average computation times versus number of targets

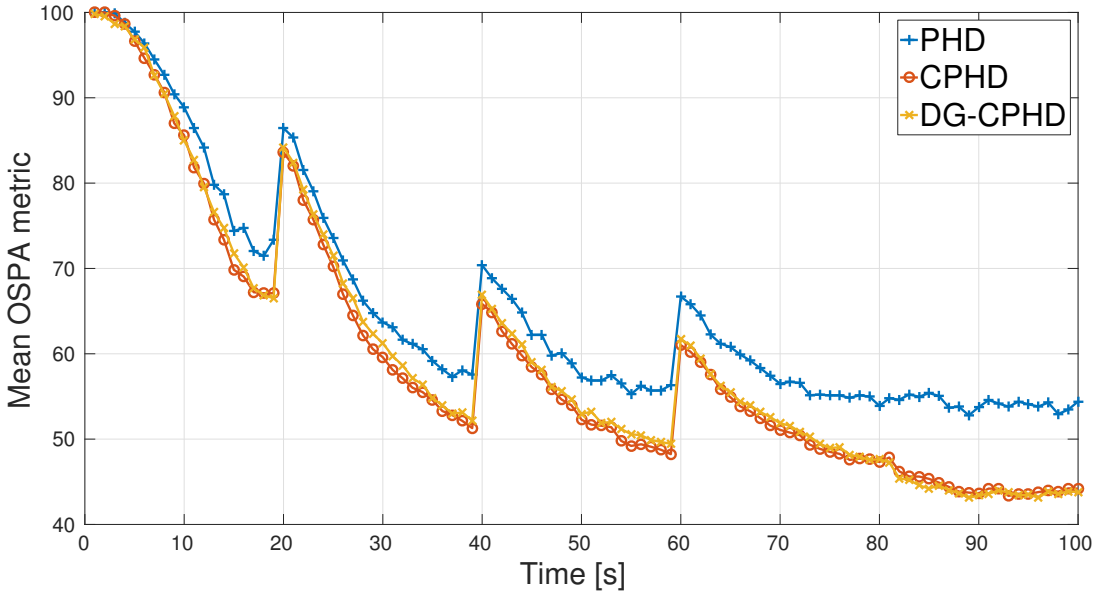


Figure 4.9: MOSPA metric over time ($p_d = 0.60$, $\lambda = 50$, $N_t = 20$)

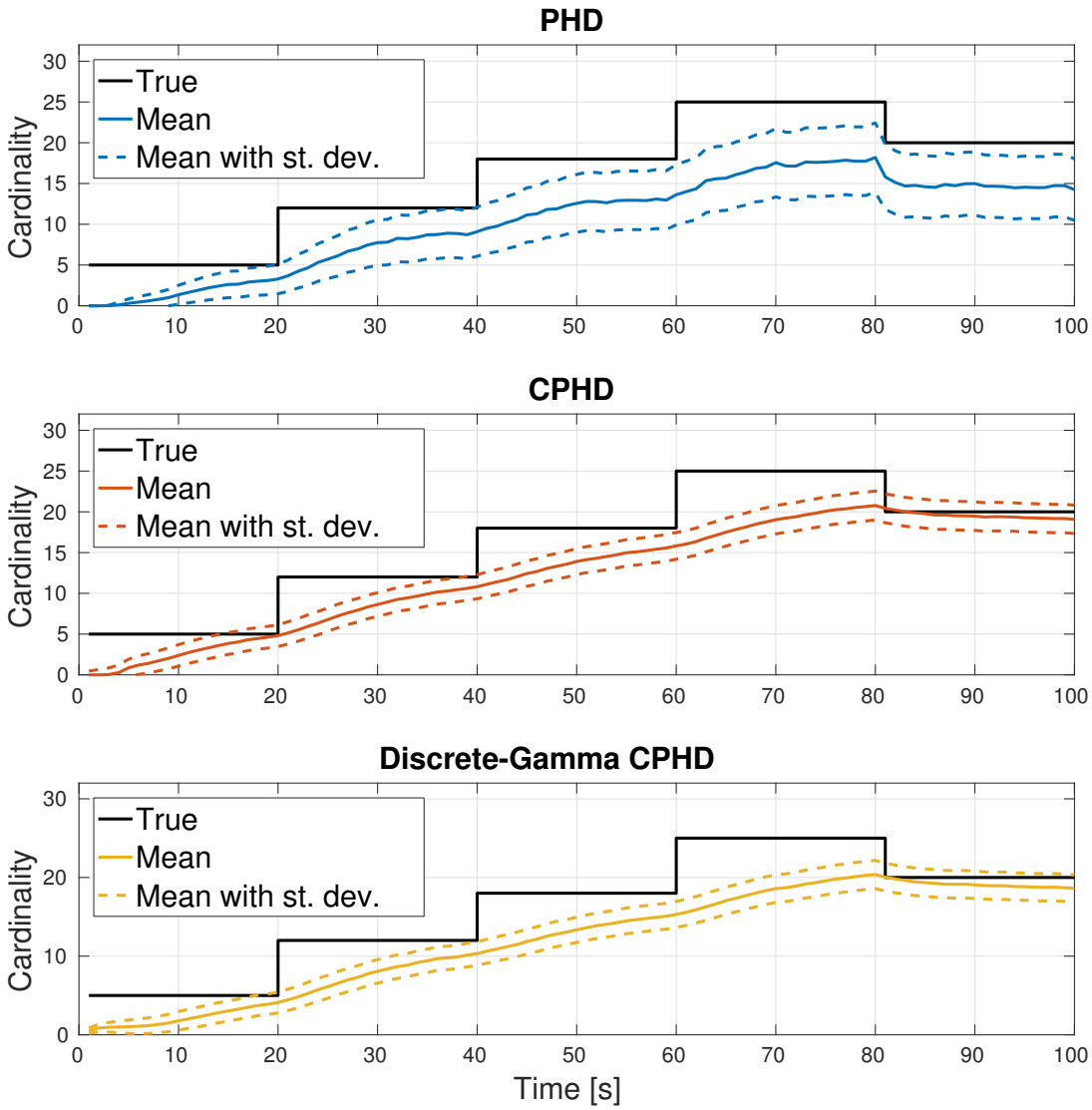


Figure 4.10: Cardinality statistics over time ($p_d = 0.60$, $\lambda = 50$, $N_t = 20$)

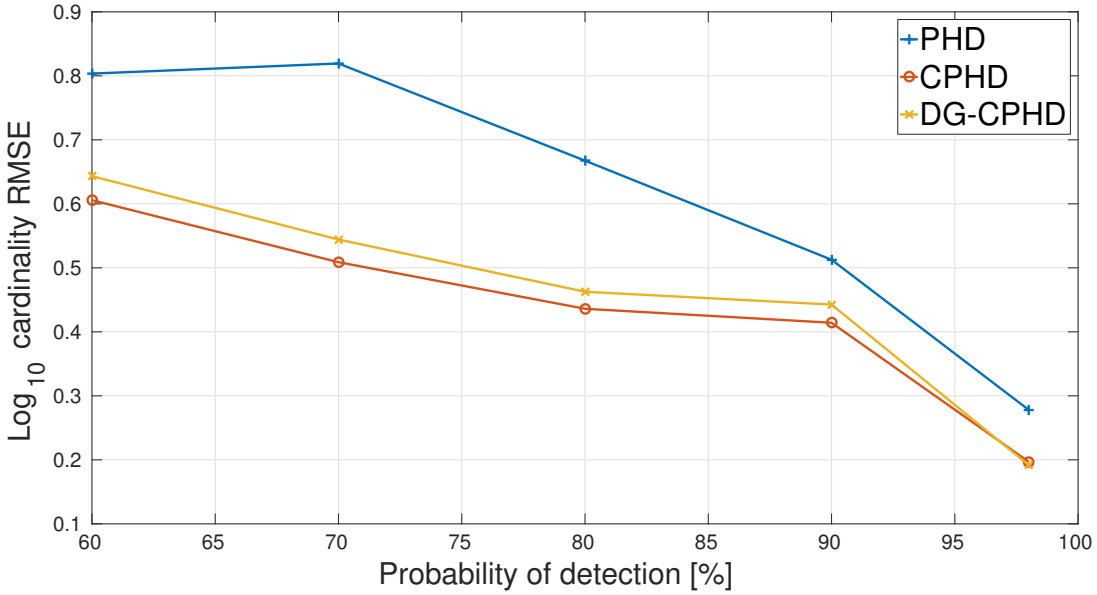


Figure 4.11: Average cardinality RMSE versus probability of detection

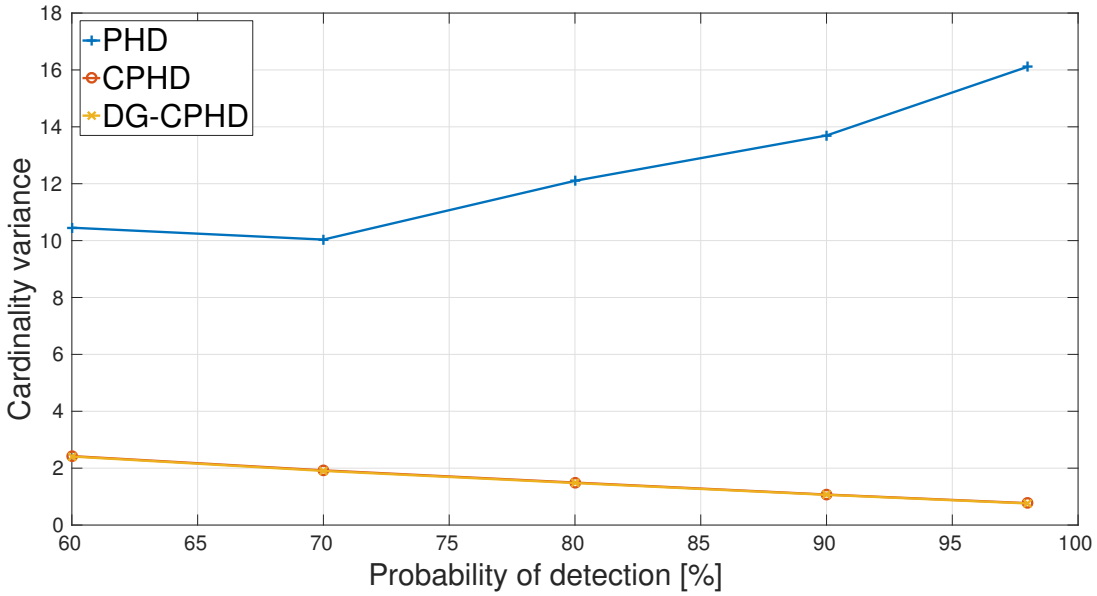


Figure 4.12: Average cardinality variance versus probability of detection

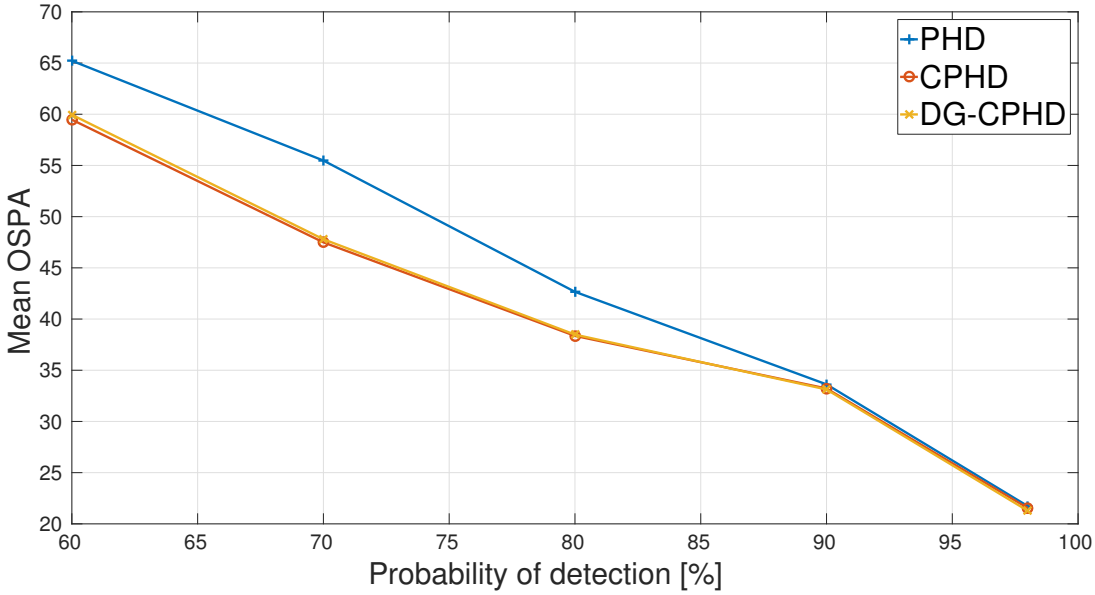


Figure 4.13: Average mean OSPA versus probability of detection

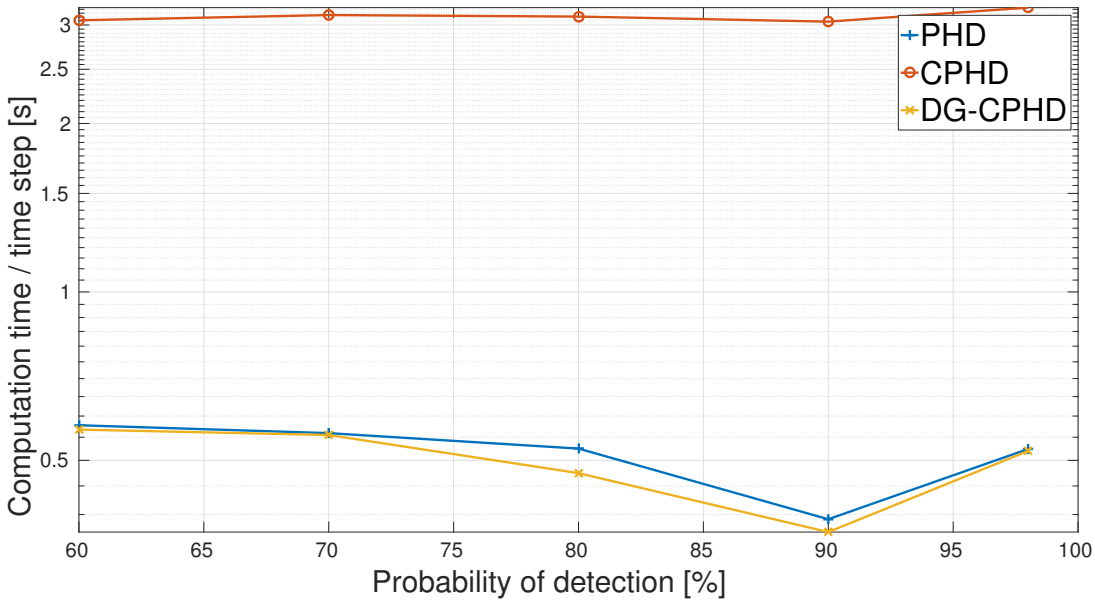


Figure 4.14: Average computation times versus probability of detection

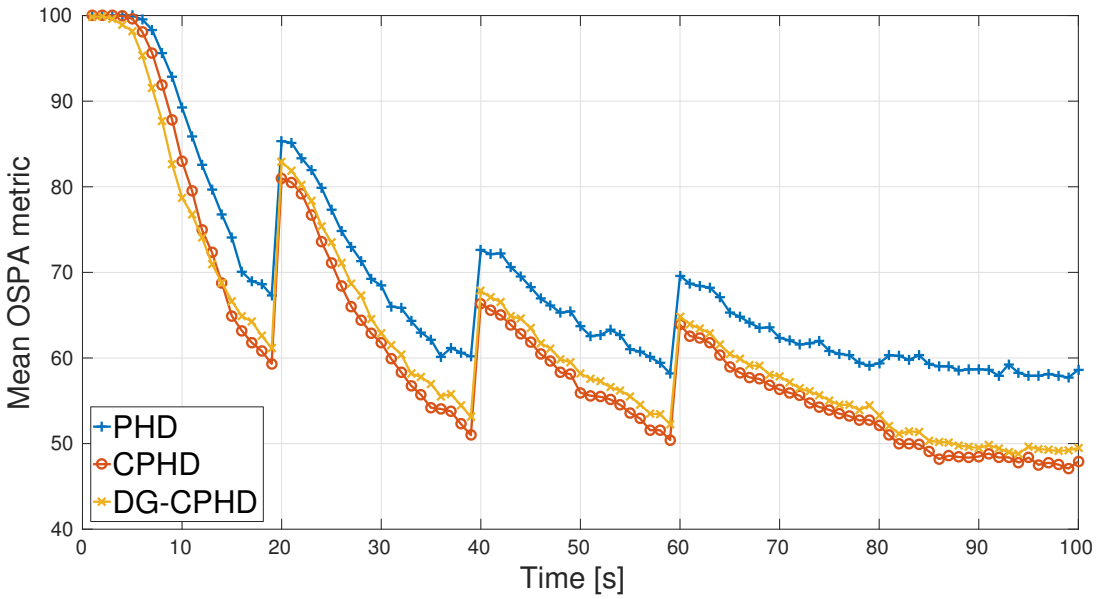


Figure 4.15: MOSPA metric over time ($p_d = 0.80$, $\lambda = 200$, $N_t = 20$)

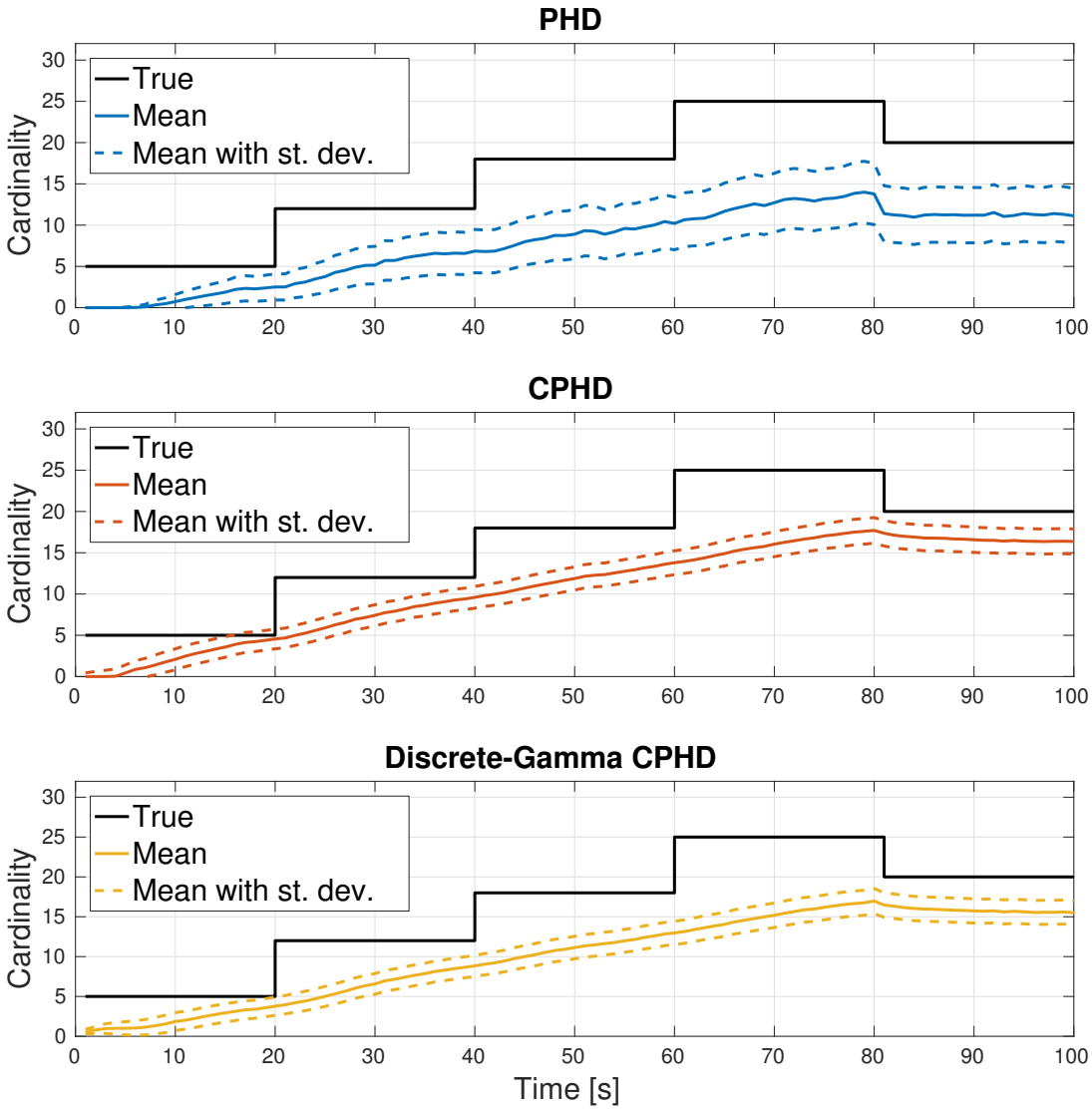


Figure 4.16: Cardinality statistics over time ($p_d = 0.80$, $\lambda = 200$, $N_t = 20$)

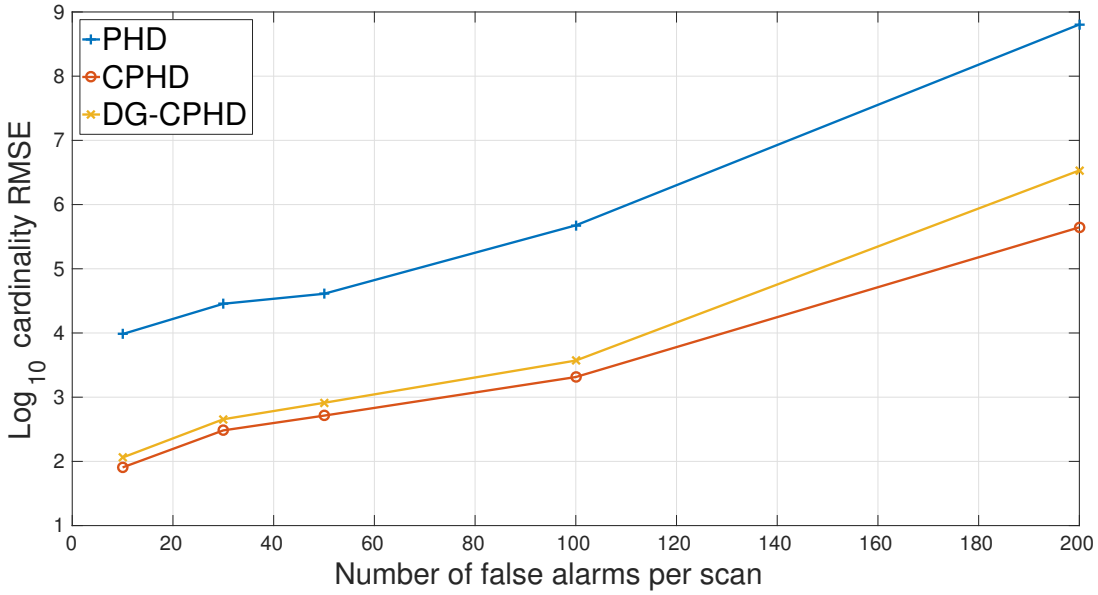


Figure 4.17: Average cardinality RMSE versus number of false alarms

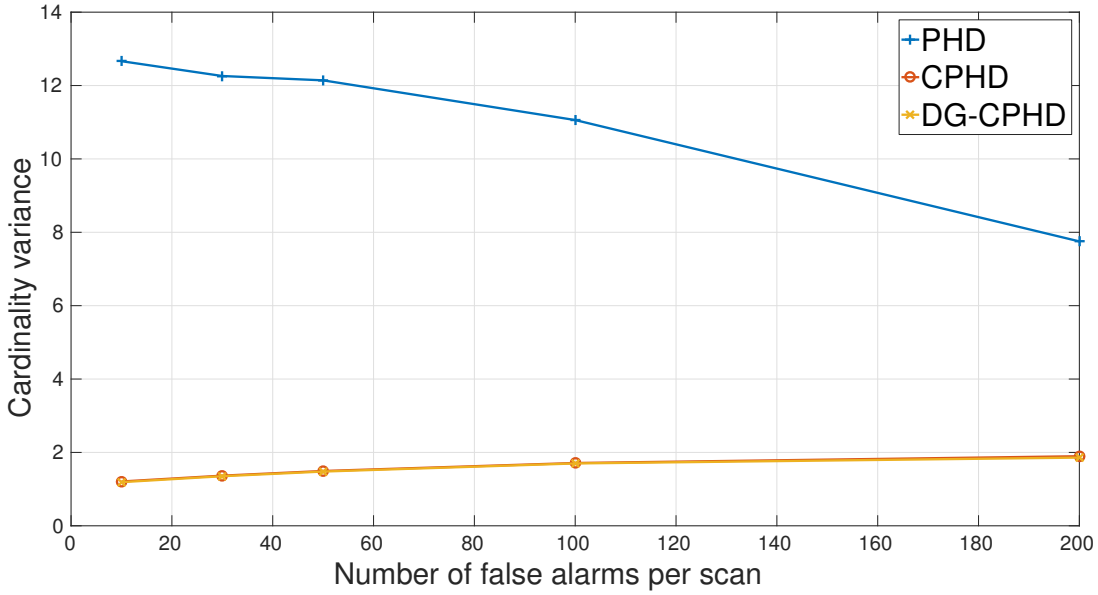


Figure 4.18: Average cardinality variance versus number of false alarms

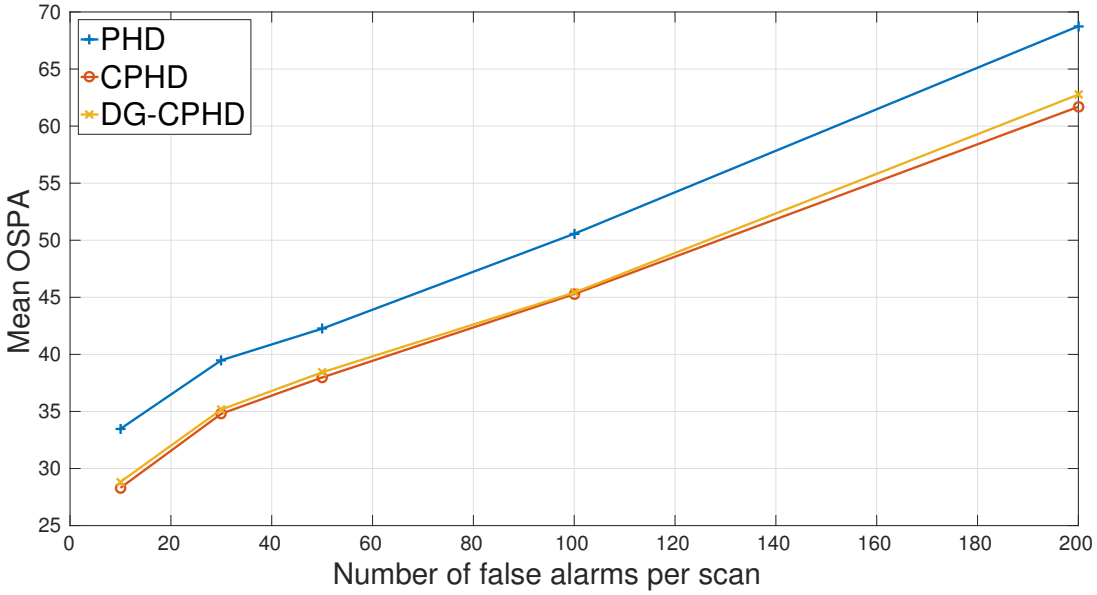


Figure 4.19: Average mean OSPA versus number of false alarms

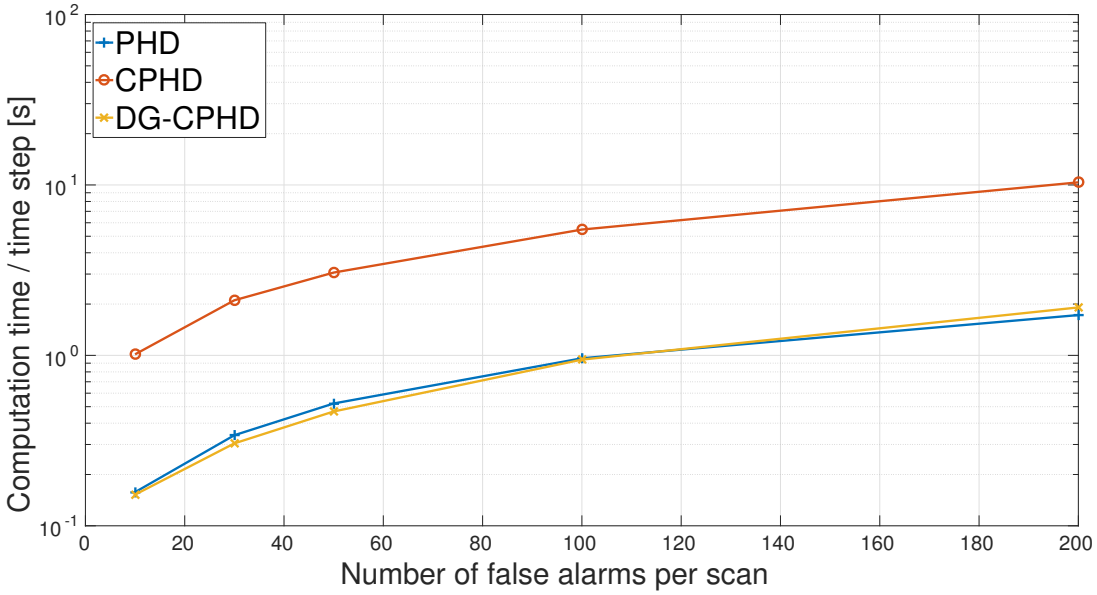


Figure 4.20: Average computation times versus number of false alarms

5

Measurement-to-measurement Association for Space Surveillance

In Chapter 0, we identified a critical challenge to multi-object estimation and data fusion: resolving data association ambiguity and representing uncertainty accurately in situations where the state observability is low. Since observability is a characteristic mostly related to the observation settings (sensor coordinate frame, observation period etc), in such situations, measurements may be well accurate and yet offer insufficient information for the state estimation to be accomplished. This is especially true in Space Surveillance applications, where short-periodic observation of geostationary satellites (and debris) by electro-optical instruments provide accurate angular measurements but offers little information about objects' orbits. This problem is very important as its solution may facilitate cataloging space debris for supporting analyses of new satellite launching, and maintaining safe operations. However, it poses many difficulties to standard methods because the association ambiguities are enormous, hundreds of objects are observed in short-term campaigns, and purely heuristic strategies (e.g., nearest neighbor) are prone to discard most of the redundant information from multiple observation sessions. In this chapter, we treat this challenge by proposing a comprehensive solution for resolving the association ambiguities and representing the uncertainty to an adequate level of accuracy. This is achieved in an extremely difficult scenario where the sensor is not able to provide angle rates, ranges, or range rates, and only provides angles.

As touched on, angles-only orbit determination of nearly geosynchronous and geostationary objects based on short-arc observations is challenging both because of their low observability and measurement uncertainty commensurate with the observed arc length. In general, the scenario requires multiple observation sessions, several hours apart, to deliver enough observability and reduced susceptibility of orbit determination to measurement noise. Ultimately, this obliges that observations from different sessions be correctly associated under the presence of other objects. In this chapter, we propose a new framework to effectively associate observations from multiple sessions for uncorrelated tracks. The proposed framework is based on a new initial orbit determination method that enables a reasonable description of nearly-geosynchronous and geostationary orbits and their uncertainties, and a procedure for statistical comparison between estimated orbits in a space comprising orbital elements and measurements. The comparison generates likelihood values that quantify the similarity of observations across sessions. The new initial orbit determination algorithm is based on Escobal's method along with estimates of

orbital elements by a new unperturbed two-body angular motion model. The resulting method is demonstrated to be computationally feasible and satisfactory in terms of performance, as evaluated by the application to a short campaign of simulated observations.

5.1 Overview

Angles-only orbit determination based on electro-optical observations constitutes a fundamental task in the context of space situational awareness (SSA), with particular importance of supporting a space surveillance and tracking (SST) framework to track and keep custody of active and decommissioned spacecraft, and debris. One major difficulty of SST activities is related to determining orbits of small objects in deep space, at nearly geostationary altitudes. This difficulty becomes even more prominent for uncorrelated track (UCT) orbits, i.e., when dealing with observations that cannot be associated with known objects. Usually, for deep space resident space objects (RSOs), surveillance data are only collected for a very small fraction of their orbit revolution, i.e., sequences of measurements cover very short arcs spanning over only a few minutes of observation per object, which results in very low observability. This scenario induces the orbit determination problem to include a range of infeasible solutions and hence demands more than one observation session to allow accurate determination of high orbits. The consequence is a need for identification of which measurements obtained in two or more sessions pertain to the same object. Ultimately, the problem turns into a challenge for quasi-geostationary objects since tens to hundreds of them may appear in each session. Additionally, the problem is further complicated by the stochastic nature of measurements, whose noise standard deviation is of a magnitude similar to the observed arc lengths, and because of numerous closely-spaced objects that exacerbate a typically large observation-to-observation associations problem.

The literature on methods for orbit determination and tracking based on short-arc angles-only observations is relatively recent. Most algorithms depend on a crucial feature: the ability to model the initial orbit uncertainty based on a locus of all possible orbits that could explain a set of measurements, called admissible region (AR). Originally proposed by Milani [139], the admissible region is established in the phase space of range versus range rate to satisfy the vis-viva equation for a given set of observed angles and angle rates. Recent methods use a constrained admissible region (CAR) to model the probability density function of an object state [60] and then start tracking the object. Provided two or more sets of observations, identifying which separate sets of data belong to the same physical object is posed as the so-called correlation problem. Notably, this problem was addressed in the context of identifying lost asteroids by Milani [138], who proposed that the orbit determination problem should be solved in two stages: 1) correlating different sets of observations, and then 2) determining the orbit to full precision. The complete procedure is called *linkage* in the literature, but this chapter will concentrate only on the first stage.

Most state-of-the-art linkage methods rely on the admissible region and are due to Tommei, Milani & Rossi [192], Farnocchia et al. [71], Gronchi, Dimare & Milani [85]. For ease of reading, we shall briefly recall the main features of these algorithms, directing the reader to the cited papers for the full mathematical treatment. Direct linkage methods propose a matching between pairs of observed arcs by: (i) propagating a grid of solutions (swarm trajectories) that outlines the admissible region [139, 192, 71] forward in time, (ii) analytical surface intersection

[134], (iii) boundary-value-based optimization [179], and (iv) direct Bayesian updates to the admissible region [74, 75]. These methods focus on how to match instances of sparsely located observations, rather than addressing the combinatorial complexity that arises in the multiple observation-to-observation association problem.

On the other hand, another class of methods appeals to multi-object Bayesian estimation techniques in the same ethos as those that comprise multiple-target tracking algorithms. These solutions start "tracklets" on sequences of measurements, with initially large estimated uncertainties modeled by the constrained admissible region, and filter them by multiple-hypothesis estimators. This family of solutions is well represented by the method called Constrained Admissible Region, Multiple Hypothesis Filter (CAR-MHF) [60, 61] and similar techniques [77, 96]. The multiple hypothesis filter (MHF) that is used is a bank of Unscented Kalman filters (UKFs) whose estimates are weighted based on the time history's statistical agreement with incoming data. Recently, new solutions have been proposed via adoption of more sophisticated filters under the same usual framework: Singh et al. [180] propose a solution based on the well known Multiple Hypothesis Tracker (MHT), Stauch et al. [186] adopt a Multiple-Hypothesis Joint Probabilistic Data Association (MH-JPDA) filter in place of the MHF, and Jones et al. [104] tackle the estimation problem by a Generalized Labeled Multi-Bernoulli (GLMB) filter.

This chapter approaches the problem from a slightly different perspective: our focus is on how to accomplish the observation-to-observation associations, working around the computational complexity required by multiple-hypothesis filtering. Our approach aims to provide a feed for a full-precision orbit determination method. Aligned with this perspective, we propose a system that avoids tracking in the first instance where, presumably, a solution in real-time is not a requirement but computational efficiency is. The proposed framework results from the combination of two fundamental techniques. The first one is a novel initial orbit determination (IOD) method that enables a reasonable description of nearly-geosynchronous and geostationary orbits and their uncertainties, by deliberately avoiding out-of-context solutions and taking into account measurement stochasticity. The second technique is a procedure for statistical comparison between estimated orbits, in a space comprising orbital elements and measurements, to generate likelihood values that quantify the similarity of observations across sessions by means of uncertainty propagation in the orbital elements space. The proposed IOD method requires a supplementary orbital motion model that simplifies computations of angular orbital elements and aids the uncertainty evaluation. This two-body unperturbed motion model is formulated in terms of spherical-trigonometric relations and, although simple, it allows computationally inexpensive and accurate enough predictions to associate observations. A performance evaluation of the whole framework is applied to a short campaign of simulated observations, demonstrating its computational feasibility and practical value, and suggesting avenues for new applications.

The outline of the chapter is as follows. The two-body unperturbed motion model is presented in Section 5.2 along with its modeling assumptions and characteristics. In Section 5.3, we present assumptions, discuss and derive the proposed IOD method for obtaining parameters for each observed arc. Formulation of uncertainty associated with the initially determined orbits is established in Section 5.4. In Section 5.5 we discuss and define the likelihood function between pairs of observed arcs and the method for determining associations across observations. In Section 5.6 we illustrate the properties of the proposed technique for a simulated campaign of

observations comprising several nights, and finally, in Section 5.7 we expose concluding remarks and expected future work.

5.2 Angular motion model

In this section, we present the angular motion model on which the IOD method is partially based. The model assumes that an object moves according to an unperturbed orbit under the sole influence of Earth's gravitational force, which consists of an idealized mathematical approximation of the actual motion. In this regime, the motion describes a Keplerian orbit that can be parameterized in many different ways, each consisting of a set of six parameters (e.g., a Keplerian orbit can be uniquely described in Cartesian coordinates by the vector $(x, y, z, \dot{x}, \dot{y}, \dot{z})^{\mathrm{T}}$).

Another orbit parameterization often used in astronomy and celestial mechanics is composed of the classical orbital elements, $(a, e, \imath, \Omega, \omega, \nu)^{\mathrm{T}}$, also known as Keplerian orbital elements, defined in Table 5.1 and partially illustrated in Figure 5.1. Commonly, as an alternative to the classical orbital elements, one can replace the true anomaly, ν, by the so-called mean anomaly, M, which describes the angular position that a fictitious object would have in a uniform circular motion with the same orbital period as that of the actual object in its elliptical orbit. In general, the mean and the true anomaly are not in phase, except at the orbit periapsis (the nearest point with respect to the Earth center) and at the apoapsis (the farthest point with respect to the Earth center). A similarly important way of describing celestial object positions is based on two angles, the right ascension and declination, which situate a body in the celestial sphere with respect to a reference direction and to the celestial equator at a given reference epoch. The right ascension, α, is the angle from the reference direction to the longitudinal position of the object in the celestial sphere, whereas the declination, δ, is the angle from the celestial equator to the latitudinal position of the object. Usually, the reference direction is set as the direction of the vernal equinox at a given reference epoch (e.g., 1st January 2000, at noon, is used to define the celestial frame known as J2000.0). The vernal equinox direction (in longitude) is defined by the event in which the plane of Earth's equator passes through the center of the Sun's disk during the northern hemisphere winter (normally on the 20th March).

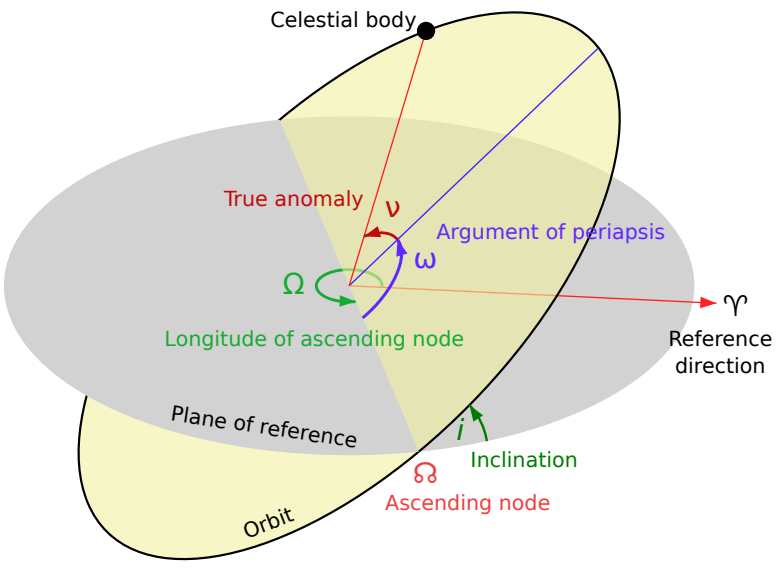
In our model, we describe the Keplerian orbits by a corresponding angular motion inscribed in the celestial sphere as depicted in Figure 5.2. The motion is characterized by the argument of latitude, defined as $u(t) \coloneqq \omega + \nu(t)$ on the orbital plane with respect to the ascending node, where ω is the argument of perigee and ν is the true anomaly. The orbital plane has inclination $\imath$ with respect to the celestial equator.

In Figure 5.2, a spherical arc covered by $u(t)$ forms a spherical right triangle[1] with arcs $\alpha(t) - \Omega$ and $\delta(t)$, where α and δ are respectively the geocentric right ascension and declination of the object, and Ω is the right ascension of the ascending node. The angle between arcs α and δ is a right angle, i.e., $\angle(\alpha, \delta) = \pi/2$. We define $\zeta \coloneqq \angle(u, \delta)$ and invoke both the spherical

[1]Note that $u(t)$ is inscribed in the celestial sphere in Figure 5.2, but its actual arc length, in the Keplerian orbit, can only be recovered if the radial position is known.

Parameter	Symbol	Description
Semi-major axis	a	Half the length of the major axis, a reference segment that is used to mathematically describe conic sections (curves). If the conic section is an ellipse, the semi-major axis is the sum of the periapsis and apoapsis distances divided by two. When the conic section is a circle, the semi-major axis is the radius of the circle.
Eccentricity	e	Scalar parameter that quantifies how much a conic section deviates from being circular. When the conic section is a circle, $e = 0$, whereas when the conic section is an ellipse $e \in (0, 1)$. The eccentricity of a parabola is $e = 1$ and that of a hyperbola is $e > 1$.
Inclination	i	Angle between the plane where the orbit is inscribed and the celestial equator (the plane defined by the Earth's equator at a reference epoch).
Right ascension of the ascending node	Ω	Angle between a reference direction (usually the vernal equinox at a given reference epoch) and the point where the celestial object crosses the celestial equator while ascending in its orbit. This crossing point is the so-called ascending node.
Argument of perigee	ω	Angle between the ascending node and the nearest point of orbit with respect to the Earth center (periapsis or perigee).
True anomaly	ν	Angle between the orbit perigee and the actual position of the object.
Mean anomaly (alternative)	M	Angle between the orbit perigee and the position that a fictitious object would have if it moved in a circular orbit, at constant angular velocity, with the same orbital period as that of the actual object in its elliptical orbit. It is an alternative parameter that replaces the true anomaly in the classical elements set.

Table 5.1: Classical orbital elements



Credit: *Wikipedia (https://en.wikipedia.org/wiki/Orbital_elements)*

Figure 5.1: Orbital elements

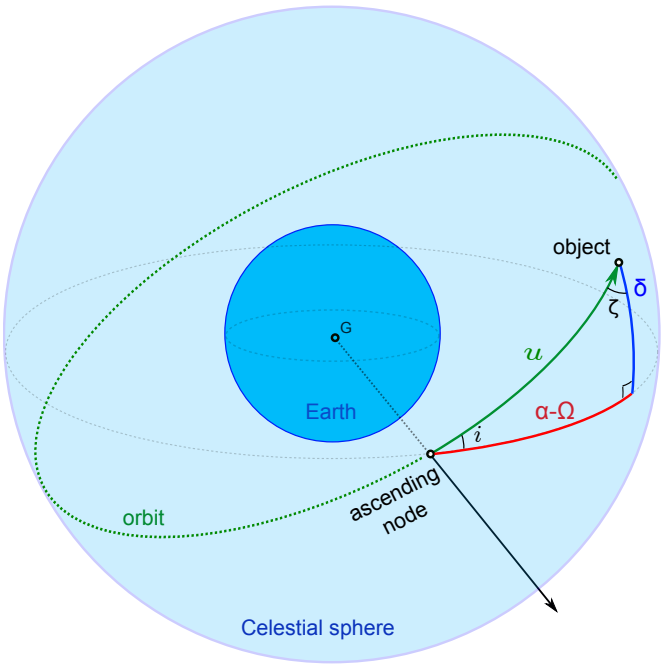


Figure 5.2: Angular motion model diagram

law of sines and law of cosines to write

$$\frac{\sin u}{\sin \frac{\pi}{2}} = \frac{\sin(\alpha - \Omega)}{\sin \zeta} = \frac{\sin \delta}{\sin i}, \tag{5.1}$$

$$\cos i = -\cos \frac{\pi}{2} \cos \zeta + \sin \frac{\pi}{2} \sin \zeta \cos \delta, \tag{5.2}$$

$$\cos u = \cos(\alpha - \Omega) \cos \delta + \sin(\alpha - \Omega) \sin \delta \cos \frac{\pi}{2}. \tag{5.3}$$

From (5.1), $\sin \zeta = \sin(\alpha - \Omega)/\sin u$, which can be substituted in (5.2) to get

$$\cos \delta = \frac{\sin u}{\sin(\alpha - \Omega)} \cos i. \tag{5.4}$$

Additionally, from (5.3),

$$\cos \delta = \frac{\cos u}{\cos(\alpha - \Omega)}. \tag{5.5}$$

Equating (5.4) and (5.5), one gets

$$\frac{\cos u}{\cos(\alpha - \Omega)} = \frac{\sin u}{\sin(\alpha - \Omega)} \cos i,$$

$$\tan(\alpha - \Omega) = \tan u \cos i. \tag{5.6}$$

Observing that $\sin \delta = \sin u \cdot \sin i$ from (5.1), and based on (5.6), the angular motion model

can be found to be described by

$$
\begin{aligned}
u(t) &= \omega + \nu(t), && (5.7)\\
\nu(t) &= M(t) + 2\sum_{k=1}^{\infty}\frac{1}{k}\left(\sum_{\ell=-\infty}^{+\infty}\beta^{|k+\ell|}\mathbb{J}_{\ell}(-ke)\right)\sin kM(t), && (5.8)\\
M(t) &= M_0 + n\cdot(t-t_0), && (5.9)\\
\alpha(t) &= \Omega + \arctan\left[\tan u(t)\cdot\cos i\right], && (5.10)\\
\delta(t) &= + \arcsin\left[\sin u(t)\cdot\sin i\right], && (5.11)\\
\lambda(t) &= \lambda_{\mathrm{GHA}} + \alpha(t) - \omega_E\cdot(t-t_0), && (5.12)\\
\phi(t) &= + \arctan\left[\tan\delta(t)/(1-e_E^2)^{1/2}\right], && (5.13)
\end{aligned}
$$

where e is the orbit eccentricity, M is the mean anomaly, M_0 is the mean anomaly at epoch t_0, n is the mean angular motion (angular velocity), λ_{GHA} is the Greenwich hour angle, ω_E is Earth's rotation velocity, e_E is Earth's first eccentricity, and λ and ϕ are respectively geodetic longitude and latitude of the ground track. Equation (5.8) is known in the literature as the equation of the center ([8], Chapter 5), where $\mathbb{J}_{\ell}(\cdot)$ is a Bessel function of first kind, of order ℓ, and $\beta = (1-\sqrt{1-e^2})e^{-1}$. The coefficients of the series $\sum_{\ell=-\infty}^{+\infty}\beta^{|k+\ell|}\mathbb{J}_{\ell}(-ke)$ in (5.8) can be developed in terms of powers of e [181, 8]. These series can be presented in truncated form and, in general, for nearly-geosynchronous orbits, eccentricities are small enough to justify approximations based on the first terms according to

$$\nu(t) = M(t) + 2e\sin M(t) + \frac{5}{4}e^2\sin 2M(t) + \mathcal{O}(e^3). \tag{5.14}$$

Where applicable in this chapter, provided a true anomaly, computing the mean anomaly requires solving Kepler's equation, which can be accomplished by the extended Newton method proposed by Markley [132] or by the efficient procedure proposed by Fukushima [76]. Equation (5.9) follows from the unperturbed motion assumption, which entails a uniform mean motion, whereas equations (5.10) and (5.11) are exactly derived from spherical-trigonometric relations (5.1), (5.2) and (5.3). The assumptions of the unperturbed motion disregard the contributions of Earth's non-spherical terms and luni-solar gravitational perturbations in the orbit. Notice that, provided the motion in the orbital plane is known and the assumptions hold (i.e., that only Earth's spherical terms are relevant), the proposed model is exact. Another important feature is that equations (5.7)–(5.11) are independent of range, which enables direct relations for angles-only orbit determination.

While it may seem that difficult problems would motivate the use of more accurate models, we advocate simplicity here on the grounds that, under severely low observability conditions, the extra parameters of a more sophisticated model would be difficult and time-consuming to estimate and would not offer significant utility (in the specific context of initial orbit determination).

5.3 Initial orbit determination

We now discuss and present the initial orbit determination method we propose for solving the observation-to-observation association problem. Before proceeding, we remind the reader of the fundamental goal of IOD: to obtain a reasonable first-order estimate of orbital elements. With that in mind, one should suspect that simplicity is also a driver for the IOD algorithm.

Classical and semi-analytical angles-only IOD methods, such as those proposed by Gauss, Laplace [194], Escobal (double range-iteration) [69], and Gooding [82], assume that the orbit must be keplerian and observations are deterministic. Apart from these assumptions, the referred IOD methods are quite general in the sense that they don't make any further assumptions about the orbit, which is a great feature to have if no prior information is available. Nevertheless, it just so happens that the counterpart of such a generality is a wide range of possible solutions, induced both by conditions of observability and measurement noise, most of which are infeasible. Classical methods are peculiarly sensitive to the observation geometry and quality. What we advocate is the other way around: we appeal to prior knowledge about objects and take into account stochasticity of observations so that most of the inappropriate solutions are deliberately discarded. While we acknowledge the consequent loss of generality our aim is to provide a good solution to the specific problem at hand.

The posed problem involves determining orbits for a subset of the population of objects residing in the so-called geostationary/geosynchronous belt. As will hopefully become clear, discarding out-of-context solutions is an essential feature of our IOD algorithm. Another cornerstone of the method is treating determined orbits as stochastic curves, whose initial randomness is solely due to measurement noise, in the exact sense of functional data analysis [158], and whose propagated uncertainty will account for the major contributions of Earth's non-spherical (and, optionally, luni-solar attraction) perturbations. Hence, under these principles, one notes that simplicity of the underlying model becomes logical since, in practical terms, the evolving uncertainties would be very likely to model the second-order effects on orbit predictions. From this perspective, the effects due to secular perturbations and periodic terms affecting the orbit are then captured by a process noise to explain effects not originally incorporated in the angular motion model (5.7)–(5.11).

5.3.1 Core of the IOD algorithm

This section describes the IOD process by referring to Figure 5.3. The core of our IOD algorithm takes as input a set of angles observed at three different time instants $\mathrm{Y}_{1:3} \coloneqq \{(\alpha_{t,k}, \delta_{t,k}),\, t_k : k = 1, 2, 3\}$, where $(\alpha_{t,k}, \delta_{t,k})$ are the topocentric right ascension and declination coordinates, and processes them to produce a first estimate of orbital elements $\tilde{\boldsymbol{\Theta}}_0 = (a, e, \imath, \Omega, \omega, M)^T$. In the vector of orbital elements, a is the semi-major axis, e is the eccentricity, Ω is the right ascension of the ascending node, ω is the argument of perigee, and M is the mean anomaly. The main procedure is underpinned by three techniques: (i) Roy's procedure [170] for simultaneous estimation of slant range and conversion of observed angles from the topocentric to geocentric frame, (ii) Escobal's formulation for estimating orbital parameters in a double range-iteration scheme [69], (iii) the angular motion model as presented in Section 5.2.

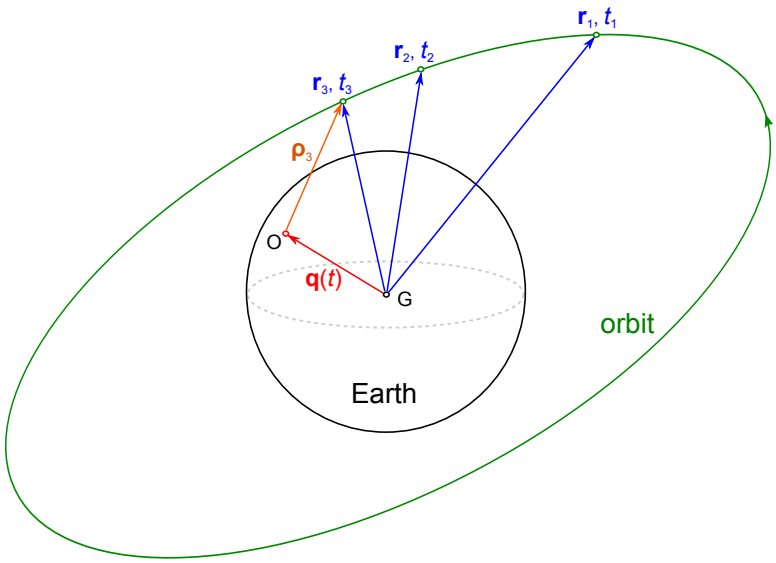


Figure 5.3: Reference diagram for the IOD problem

Roy's procedure

Archie Roy devised a method for conversion of observed angles in the topocentric coordinate frame, (α_t, δ_t), based on the observation site position vector $\mathbf{q}$, to obtain angles in the geocentric frame, (α, δ), along with the associated slant range ρ (see [170], pages 66-67). Roy's procedure is based on the series of equations

$$\tan(\alpha - \alpha_t) = \frac{(q/\rho)\cos\phi \cdot \sin(\alpha_t - \lambda_{\text{LST}})}{\cos\delta_t + (q/\rho)\cos\phi \cdot \cos(\alpha_t - \lambda_{\text{LST}})}, \tag{5.15}$$

$$\tan\varepsilon = \frac{\tan\phi \cdot \cos\left(\frac{\alpha_t - \alpha}{2}\right)}{\cos\left(\frac{\alpha_t + \alpha}{2} - \lambda_{\text{LST}}\right)}, \tag{5.16}$$

$$\tan(\delta_t - \delta) = \frac{(q/\rho)\sin\phi \cdot \sin(\delta_t - \varepsilon)}{\sin\varepsilon + (q/\rho)\cos\phi \cdot \sin(\delta_t - \varepsilon)}, \tag{5.17}$$

$$\rho = \sqrt{r^2 + q^2 - 2r(\hat{\mathbf{r}} \cdot \mathbf{q})}, \tag{5.18}$$

where $\hat{\mathbf{r}} = (\cos\alpha{\cdot}\cos\delta, \sin\alpha{\cdot}\cos\delta, \sin\delta)^T$, ε is an auxiliary variable, and λ_{LST} is the local sidereal time. Given an accurate estimate for the geocentric radius, r, the parameters (α, δ, ρ) can be obtained with very few iterations of equations (5.15)–(5.18). We remark that for computing the (J2000.0) mean sidereal time, required by Roy's procedure, we use the fairly accurate methods by the U.S. Naval Observatory Vector Astrometry Software, C Edition, version 3.1 (NOVAS C3.1)[2], incorporating Earth's precession and nutation, and historical data of celestial polar deviations.

Escobal's formulation

The second technique on which the IOD procedure relies is Escobal's formulation in a double range-iteration scheme. Actually, our IOD algorithm does not follow Escobal's method, i.e., it does not iterate a pair of geocentric radii from distinct points in the observed arc, but rather makes computations based on his formulation. The crucial part relates to what is

[2] Available at http://aa.usno.navy.mil/software/novas/novas_info.php.

called Gibb's problem, i.e., given three known positions $\{r_k : k = 1, 2, 3\}$ on three equations $p \coloneqq a(1 - e^2) = r_k(1 + e\cos\nu)$, determine the three orbital parameters (a, e, ν). These three equations can be combined to produce the symmetric formula

$$p = \frac{\sin\Delta\nu_{32} - \sin\Delta\nu_{31} + \sin\Delta\nu_{21}}{r_1^{-1}\sin\Delta\nu_{32} - r_2^{-1}\sin\Delta\nu_{31} + r_3^{-1}\sin\Delta\nu_{21}}, \tag{5.19}$$

where the terms $\Delta\nu_{ij} = \nu_i - \nu_j$, $i \neq j$, can be computed from estimates of $\{\mathbf{r}_k : k = 1, 2, 3\}$ via relations

$$\cos\Delta\nu_{ij} = \frac{\mathbf{r}_i \cdot \mathbf{r}_j}{r_i r_j}, \tag{5.20}$$

$$\sin\Delta\nu_{ij} = \frac{\|\mathbf{r}_i \times \mathbf{r}_j\|_2}{r_i r_j}, \quad (i, j) = (3, 2), (3, 1), (2, 1). \tag{5.21}$$

Escobal's formulation identifies a well defined way to estimate the semi-latus rectum as

$$p = \begin{cases} \dfrac{+c_1 r_1 + c_3 r_3 - r_2}{+c_1 + c_3 - 1}, & \Delta\nu_{31} > \pi; \\ \dfrac{-c_2' r_2 + c_3' r_3 + r_1}{-c_2' + c_3' + 1}, & \Delta\nu_{31} \leq \pi; \end{cases} \tag{5.22}$$

where

$$c_1 = \frac{r_2}{r_1}\frac{\sin\Delta\nu_{32}}{\sin\Delta\nu_{31}}, \quad c_3 = \frac{r_2}{r_3}\frac{\sin\Delta\nu_{21}}{\sin\Delta\nu_{31}}, \tag{5.23}$$

$$c_2' = \frac{r_1}{r_2}\frac{\sin\Delta\nu_{31}}{\sin\Delta\nu_{32}}, \quad c_3' = \frac{r_1}{r_3}\frac{\sin\Delta\nu_{21}}{\sin\Delta\nu_{32}}. \tag{5.24}$$

From estimate (5.22), well defined values can be found for

$$e\cos\nu_k = \frac{p}{r_k} - 1, \quad k = 1, 2, 3. \tag{5.25}$$

In addition

$$e\sin\nu_2 = \begin{cases} \frac{-\cos\Delta\nu_{21}\cdot e\cos\nu_2 + e\cos\nu_1}{\sin\Delta\nu_{21}}, & \Delta\nu_{31} \neq \pi; \\ \frac{+\cos\Delta\nu_{32}\cdot e\cos\nu_2 - e\cos\nu_3}{\sin\Delta\nu_{32}}, & \Delta\nu_{31} = \pi; \end{cases} \tag{5.26}$$

from which it is possible to compute

$$e = \sqrt{(e\sin\nu_2)^2 + (e\cos\nu_2)^2}, \tag{5.27}$$

$$a = \frac{p}{1 - e^2}, \tag{5.28}$$

$$n = \sqrt{\mu/a^3}, \tag{5.29}$$

$$\nu_2 = \arctan\left[\frac{e\sin\nu_2}{e\cos\nu_2}\right]. \tag{5.30}$$

The middle eccentric anomaly, middle mean anomaly, and first extremal mean anomaly are

evaluated according to the well known expressions [194]

$$E_2 = 2\arctan\left[\frac{\sqrt{1-e}}{\sqrt{1+e}}\tan\left(\frac{\nu_2}{2}\right)\right], \tag{5.31}$$

$$M_2 = E_2 - e\cdot\sin E_2, \tag{5.32}$$

$$M_1 = M_2 - n(t_2 - t_1). \tag{5.33}$$

The orbit mean anomaly is set as $M \coloneqq M_1$. Note that the mean anomaly, M, is referred to the first extremal, at epoch t_1. This is such that the observation start establishes a time reference for predictions.

Estimates from the angular motion model

The motion model presented in Section 5.2 enables computation of the remaining orbital elements. Computing the right ascension of the ascending node requires care since the whole procedure is sensitive to its value. Combining relations (5.1) and (5.2), one obtains

$$\tan i = \frac{\sin i}{\cos i} = \frac{\sin\zeta\cdot\frac{\sin\delta}{\sin(\alpha-\Omega)}}{\sin\zeta\cdot\cos\delta} = \frac{\tan\delta}{\sin(\alpha-\Omega)}. \tag{5.34}$$

By making $\tan i_1 = \tan i_2 = \tan i_3$, we write

$$\frac{\tan\delta_1}{\sin(\alpha_1-\Omega)} = \frac{\tan\delta_2}{\sin(\alpha_2-\Omega)} = \frac{\tan\delta_3}{\sin(\alpha_3-\Omega)}, \tag{5.35}$$

which for any of the pairs $(i,j) = (1,2),(1,3),(2,3)$, results

$$\tan\Omega_{ij} = \frac{\tan\delta_i\cdot\cos\alpha_j - \tan\delta_j\cdot\cos\alpha_i}{\tan\delta_i\cdot\sin\alpha_j - \tan\delta_j\cdot\sin\alpha_i}. \tag{5.36}$$

For computing right ascension of the ascending node, we set a first estimate $\Omega_0 \coloneqq \Omega_{13}$ and then iterate it using the Newton-Raphson method to minimize the cost function

$$\begin{aligned}\ell = &\left[\tan\delta_1\sin(\alpha_2-\Omega) - \tan\delta_2\sin(\alpha_1-\Omega)\right]^2 \\ &+ \left[\tan\delta_1\sin(\alpha_3-\Omega) - \tan\delta_3\sin(\alpha_1-\Omega)\right]^2 \\ &+ \left[\tan\delta_2\sin(\alpha_3-\Omega) - \tan\delta_3\sin(\alpha_2-\Omega)\right]^2.\end{aligned} \tag{5.37}$$

Only a few iterations $\Omega_{n+1} \leftarrow \Omega_n - (\{(\Omega_n)/\{'(\Omega_n))$ are needed for convergence. Notice that when $\ell = 0$, the relations given by (5.35) hold. The resulting Ω must be identified in the correct quadrant: for instance, if $\delta_2 > 0$ and $\alpha_2 - \Omega > \pi$, then $\Omega \leftarrow \Omega + \pi$ because for $0 < i_2 < \pi/2$ a positive declination necessarily means $\alpha_2 - \Omega$ should be either in the first ($\delta_3 - \delta_1 > 0$) or second ($\delta_3 - \delta_1 < 0$) quadrants. Orbit inclinations can be evaluated directly from equation (5.34):

$$i_k = \arctan\left[\frac{\tan\delta_k}{\sin(\alpha_k-\Omega)}\right], \quad k = 1,2,3. \tag{5.38}$$

We set the orbit inclination angle to be the one computed for the middle point, i.e., $i \coloneqq i_2$.

The mean argument of latitude can be calculated as follows

$$u_k = \begin{cases} \arcsin\left[\frac{\sin\delta_k}{\sin i_k}\right], & |\sin i_k| \geq |\sin\delta_k|\,; \\ \arctan\left[\frac{\tan\delta_k \sec(\alpha_k - \Omega)}{\sin i_k}\right], & \text{otherwise.} \end{cases} \tag{5.39}$$

Clearly, if $u_3 - u_1 < 0$, then the mean argument of latitude was identified in the wrong quadrant, i.e., $u_k \leftarrow \pi - u_k$. Finally, the argument of perigee is calculated by

$$\omega \coloneqq \omega_2 = u_2 - \nu_2. \tag{5.40}$$

Outline of the main IOD procedure

The main IOD routine is briefly outlined as follows

1. Set semi-major axis a and mean angular motion n as for geostationary/geosynchronous objects.
2. Compute first estimates for slant-ranges $\{\rho_k : k = 1, 2, 3\}$ using Roy's procedure (Section 5.3.1) for a geostationary radius.
3. Update of geocentric position vectors by

$$\mathbf{r}_k = \mathbf{q}_k + \rho_k \hat{\boldsymbol{\rho}}_k, \quad k = 1, 2, 3. \tag{5.41}$$

4. Using Escobal's formulation (Section 5.3.1), compute
   (a) semi-latus rectum p,
   (b) eccentricity e,
   (c) the update of semi-major axis a and mean motion n,
   (d) middle true anomaly ν_2.
5. Compute middle eccentric anomaly E_2, middle mean anomaly M_2, and first extremal mean anomaly M_1 (Section 5.3.1).
6. Transform observed angles from topocentric to geocentric frame to give $\{(\alpha_k, \delta_k) : k = 1, 2, 3\}$ and update slant-ranges $\{\rho_k : k = 1, 2, 3\}$ by using Roy's procedure.
7. Using equations derived from the angular motion model (Section 5.3.1), estimate
   (a) right ascension of the ascending node Ω,
   (b) orbit inclination i,
   (c) middle argument of latitude u_2,
   (d) argument of perigee ω.
8. Based on $\tilde{\boldsymbol{\Theta}}_0 = (a, e, i, \Omega, \omega, M)^T$, compute estimates of topocentric angles $\{(\hat{\alpha}_{t,k}, \hat{\delta}_{t,k}) : k = 1, 2, 3\}$ and verify the squared deviation

$$\epsilon^2 = \sum_{k=1}^{3} \left\| (\hat{\alpha}_{t,k}, \hat{\delta}_{t,k}) - (\alpha_{t,k}, \delta_{t,k}) \right\|_2^2. \tag{5.42}$$

If ϵ is greater than a prescribed value, re-process from Step 3 with the new values of a, n, and $\{\rho_k : k = 1, 2, 3\}$; else output $\tilde{\mathbf{\Theta}}_0$.

Single-measurement case

In some cases, a single measurement is provided for an observed object. In those cases, we make further assumptions to fix parameters for tentatively describing an orbit, although it might not be very close to the actual motion. We rely on the fact that, in general, a single measurement is obtained for objects that are very close to geostationary, since during observation no relative motion between the object and Earth's rotational phase has been perceived. The semi-major axis a and mean angular motion n are fixed as of geostationary/geosynchronous objects. Then estimates of slant-range and geocentric angles $(\rho_1, \alpha_1, \delta_1)$ are computed using Roy's procedure for a geostationary radius. Eccentricity is fixed at $e = 0$ and inclination is set at the absolute value of geocentric declination $\imath = |\delta_1|$. These assumptions imply that the orbit is circular and so the argument of perigee could be set to any reference value: we set $\omega \coloneqq 0$. If $\delta_1 \geq 0$, then u should be in one of the first two quadrants, thus $u = \omega + \nu = M$ and Ω must be chosen as pairs. One trivial possibility is $\alpha_1 - \Omega = M = \pi/2$ for $\delta_1 \geq 0$ and $\alpha_1 - \Omega = M = 3\pi/2$ otherwise. Notice that, for $\delta_1 \geq 0$, $\imath = \delta_1$, $\Omega = \alpha_1 - \frac{\pi}{2}$, $M = \frac{\pi}{2}$, and so equations (5.10) and (5.11) of the model are valid

$$\begin{aligned} \alpha_1 - \Omega &= \arctan\left[\tan(\omega + M) \cdot \cos\imath\right], \\ \frac{\pi}{2} &= \arctan\left[\tan\frac{\pi}{2} \cdot \cos\imath\right], \end{aligned}$$

$$\begin{aligned} \delta_1 &= \arcsin\left[\sin\left(\omega + M\right) \cdot \sin\imath\right], \\ \delta_1 &= \arcsin\left[\sin\frac{\pi}{2} \cdot \sin\imath\right], \end{aligned}$$

where equalities hold since $\tan\frac{\pi}{2} \to +\infty$ and $\sin\frac{\pi}{2} = 1$. These are also valid for $\delta_1 < 0$, with $\imath = -\delta_1$, $\Omega = \alpha_1 - \frac{3\pi}{2}$, and $M = \frac{3\pi}{2}$.

5.4 Treatment of initial orbit uncertainty

An adequate treatment of uncertainty associated with the orbital elements is crucial, both because one would like to explain a range of possible orbits in a sensible way and because a stochastic description is explicitly required for predictions that propagate uncertainty. The admissible region [139] is perhaps the most comprehensive way of modeling uncertainty, as it encompasses the whole universe of possible solutions. However, for the purpose of associating objects in the geostationary and geosynchronous context, we are only interested in the uncertainty within that context.

Let the "complete" uncertainty of orbital elements to be described by a probability density function $p(\mathbf{\Theta})$. One could think of the uncertainty we want to model as a conditional one, i.e., $p(\mathbf{\Theta}|a \in \mathcal{A}_{\text{geo}})$, where $\mathcal{A}_{\text{geo}}$ is the set of possible semi-major axes of objects in the GEO belt. The prior information $(a \in \mathcal{A}_{\text{geo}})$ rules out out-of-context possibilities in the statistical description. Indeed, this conditional model is appropriate for comparison between tentative objects since the same prior information affects them all equivalently. However, it may not

provide suitable predictions in absolute terms, i.e., it may not quantify the "actual" uncertainty related to an object's orbit.

Another perspective on our treatment of uncertainty is in the sense of functional data analysis (FDA) [158]. This becomes evident when one notices that Keplerian orbits delineate smooth curves, for which stochastic realization can be presumed to be due to measurement noise. The essence of this view entails that what really matters is how certain we are about the orbit in which one object might be, rather than about its actual position. In this representation, an observed arc can be modeled by a stochastic parametric process in the observational space, i.e., a process describing random parametric curves that might have generated a sequence of observations.

Suppose that all measurements have been translated to the geocentric coordinate frame using Roy's procedure for an assumed radial position. Let $\gamma(t) \coloneqq (\alpha(t), \delta(t))$ be a single parametric curve that could delineate a deterministic arc had the exact values of right ascension and declination been known without noise. We write the stochastic parametric process $\{\gamma(t)\}_{t\geq t_0}$, defining a family of random, smooth parametric curves. Functional data analysis operates on a space $\mathbb{L}^2$, usually, the Hilbert space, of square Lebesgue-integrable functions corresponding to realizations $\gamma_1(t), \gamma_2(t), \ldots, \gamma_N(t)$ of the functional process. Although $\{\gamma(t)\}_{t\geq t_0}$ is actually defined by both processes $\{\alpha(t)\}_{t\geq t_0}$ and $\{\delta(t)\}_{t\geq t_0}$, we abstract it as a single function of t that jointly explains the behavior of α and δ. Alternatively, this could be done explicitly by setting the process as the arc length $\gamma(t) = \int_{t_0}^{t} \sqrt{1 - (d\delta/d\alpha)^2}\dot{\alpha}dt$ for instance.

Each of the realizations could be seen as an infinite-dimensional vector

$$\gamma_i(t) \equiv (\gamma_i(t_0), \ldots, \gamma_i(t_1), \ldots, \gamma_i(t))^T, \quad t_0 < t_1 < \cdots < t,\, t \in \mathbb{R}_+,$$

i.e., there may be infinite many values of t (and γ_i) within each interval $[t_0, t_1]$, $[t_1, t_2]$, etc. When these realizations are defined on a finite and discretized interval $I = \{t_k : k = 0, 1, \ldots, K\}$, they are nothing but multi-dimensional vectors $\gamma_i(t_K) \equiv (\gamma_i(t_0), \gamma_i(t_1), \ldots, \gamma_i(t_K))^T$, $K \in \mathbb{N}$. Statistics can be formally taken for the functional process $\{\gamma(t)\}_{t\geq t_0}$ to obtain a mean parametric curve $\mu_{\alpha,\delta}(t) = \mathbb{E}[\gamma(t)]$ and a functional covariance $\Sigma_{\alpha,\delta}(t, s) = \mathrm{Cov}[\gamma(t), \gamma(s)]$. In the interval I these statistics correspond to the usual multi-dimensional statistics.

We recall the assumption that randomness of the functional process arises purely due to the measurement uncertainty, herein modeled by the Gaussian noise

$$\mathbf{v}_k \sim \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \sigma^2_{\alpha,m} & 0 \\ 0 & \sigma^2_{\delta,m} \end{pmatrix}\right). \tag{5.43}$$

In addition, because noise on each observed pair (α_k, δ_k) is assumed independent of other observations in a sequence of measurements, the functional covariance should have null off-diagonal terms, i.e.,

$$\Sigma_{\alpha,\delta}(t_k, t_k) \coloneqq \sigma^2_{\alpha,\delta}(t_k), \quad \Sigma_{\alpha,\delta}(t_k, t_l) = 0,\, k \neq l. \tag{5.44}$$

Clearly, the functional variances $\sigma^2_{\alpha,\delta}(t_k)$ correspond to the effect of measurement noise variances

on each time sample from the family of random curves, i.e.,

$$\begin{aligned}\sigma^2_{\alpha,\delta}(t_k) &\coloneqq \sigma^2_{\gamma,m}(t_k)\\ &= \left(\frac{\partial\gamma}{\partial\alpha}(t_k)\right)^2 \sigma^2_{\alpha,m} + \left(\frac{\partial\gamma}{\partial\delta}(t_k)\right)^2 \sigma^2_{\delta,m}.\end{aligned} \tag{5.45}$$

Had γ been analytically described beforehand, an explicit representation of functional statistics would be possible by expansion in series of orthogonal functions that form an orthonormal basis in $\mathbb{L}^2(I)$, the so-called polynomial chaos expansions. However, γ is implicitly defined in terms of unknown functions $(\alpha(t), \delta(t))$, and certainly, there are not enough realizations of curves to enable empirical computation of a mean curve and functional covariance. Despite these difficulties, if $\{\gamma(t)\}_{t\geq t_0}$ is assumed normal in the functional space, one can write

$$\mu_{\alpha,\delta}(t) = \inf_{\gamma\in\mathbb{L}^2([t_0,t])} \sum_{k=0}^{K} \|\gamma(t_k) - (\alpha_k, \delta_k)\|_2^2\,, \tag{5.46}$$

$$\sigma_{\alpha,\delta}(t) = \left\{\sigma(t) : \Pr\left[\frac{K\mathfrak{d}^2}{q^{(K)}_{c/2}} < \sigma^2(t) < \frac{K\mathfrak{d}^2}{q^{(K)}_{1-c/2}}\right] = 1 - c\right\}, \tag{5.47}$$

$$\mathfrak{d}^2 = \frac{1}{K}\sum_{k=0}^{K} \|(\alpha_k, \delta_k) - \mu_{\alpha,\delta}(t_k)\|_2^2\,, \tag{5.48}$$

where $q^{(K)}_p$ is the p-th quantile of the chi-square distribution with K degrees of freedom, and $1 - c$ is the confidence level. Smoothing and fitting techniques are prevalent in FDA, and provide a way of implementing (5.46) and (5.47). We pick a simple hypothetical curve $\gamma_h(t) \coloneqq (\delta_h(\alpha(t), \boldsymbol{\beta}), \alpha(t))$, where $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_L)^T$ are the coefficients, and fit it to the data $\{(\alpha_k, \delta_k) : k = 0, 1, \ldots, K\}$ to result $\mu_{\alpha,\delta}(t) = (\delta_h(\alpha(t), \bar{\boldsymbol{\beta}}), \alpha(t))$. Notice that a straight line $\delta_h(\alpha(t), \boldsymbol{\beta}) = \beta_0 + \beta_1\alpha(t)$ is a suitable hypothesis since at ranges around $r = 42,164.137\,\text{km}$ arcs show a curvature too small to be perceived within a length covering only $\Delta\alpha \approx 0.5°$: the typical deviation from a circular orbit would be around $\Delta r \approx 3.8 \cdot 10^{-5} r \approx 1.6\,\text{km}$. Then, we compute confidence bounds on the fitted coefficients $\bar{\boldsymbol{\beta}}$, defining a $(L+1)$-cuboid $[\bar{\boldsymbol{\beta}} - \Delta\boldsymbol{\beta}, \bar{\boldsymbol{\beta}} + \Delta\boldsymbol{\beta}]$, for a confidence level $1 - c = 99.73\%$, i.e., $\Delta\beta_l \approx 3\sigma_{\beta_l}$ for $l = 0, \ldots, L$. Using a QR decomposition of the Jacobian matrix

$$\begin{aligned}\mathbf{J} &= [\mathrm{J}_{kl}] = \mathbf{Q}\cdot\mathbf{R},\\ \mathrm{J}_{kl} &= \frac{\partial\delta_h(\alpha_k, \boldsymbol{\beta})}{\partial\beta_l},\ k = 0, \ldots, K,\ l = 0, \ldots, L;\end{aligned}$$

the coefficient bounds are given by[3]

$$\begin{aligned}\mathbf{B} &= (\mathbf{R}^T\mathbf{R})^{-1}(K-1)^{-1}\sum_{k=0}^{K}(K+1)^{-1}(\delta_h(\alpha_k, \boldsymbol{\beta}) - \delta_k)^2,\\ \Delta\boldsymbol{\beta} &= -P_t^{-1}\left(1 - c/2, K - L\right)(\mathrm{B}_{00}^{1/2}, \ldots, \mathrm{B}_{LL}^{1/2})^T,\end{aligned} \tag{5.49}$$

where $P_t^{-1}(\xi, \kappa)$ is the Student-T inverse cumulative distribution function for level ξ and κ

[3]Similar to the method provided by MATLAB for confidence intervals of fitting coefficients.

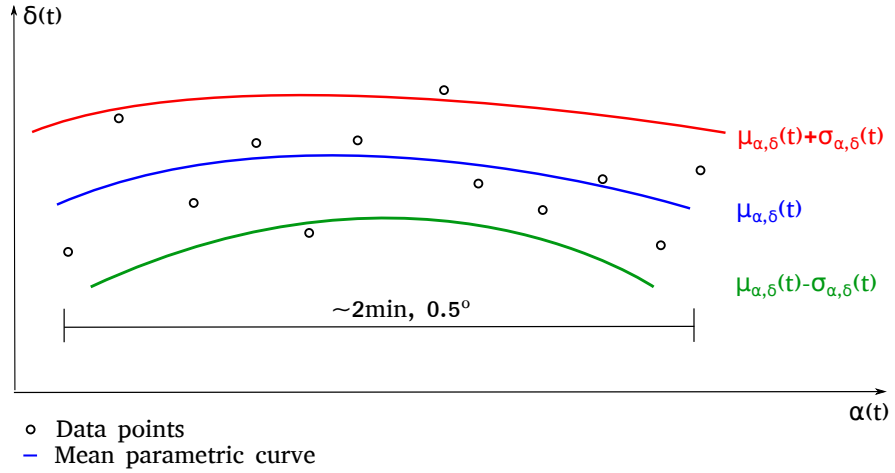


Figure 5.4: Statistical representation of the functional curves

degrees of freedom. The Student-T distribution is suitable when computing confidence intervals for estimates based on a small number of samples (measurements) and the population (co)variance is unknown. From the confidence hypercuboid $[\bar{\boldsymbol{\beta}} - \Delta\boldsymbol{\beta}, \bar{\boldsymbol{\beta}} + \Delta\boldsymbol{\beta}]$, we select the pair of vertices in a diagonal producing the most dissimilar extremal curves. These vertices define bounds of the region $[\mu_{\alpha,\delta}(t) - 3\sigma_{\alpha,\delta}(t), \mu_{\alpha,\delta}(t) + 3\sigma_{\alpha,\delta}(t)]$, valid for $t \in I$. For instance, suppose vertices $V_a = (\bar{\beta}_0 - \Delta\beta_0, \bar{\beta}_1 + \Delta\beta_1)$ and $V_b = (\bar{\beta}_0 + \Delta\beta_0, \bar{\beta}_1 - \Delta\beta_1)$ produce the most dissimilar extremal parametric curves, then V_a gives the coefficients to delineate $\mu_{\alpha,\delta}(t) - 3\sigma_{\alpha,\delta}(t)$ and V_b the coefficients for $\mu_{\alpha,\delta}(t) + 3\sigma_{\alpha,\delta}(t)$. Dissimilarity between extremal parametric curves is evaluated in terms of direction, i.e., two curves γ_1 and γ_2 are similar if their inner product $\langle \gamma_1, \gamma_2 \rangle = 1$ and are increasingly dissimilar as $\langle \gamma_1, \gamma_2 \rangle \to 0$.

Ultimately, the functional uncertainty is expressed by three curves in the observational space, $\mu_{\alpha,\delta}(t)$, $\mu_{\alpha,\delta}(t) - 3\sigma_{\alpha,\delta}(t)$ and $\mu_{\alpha,\delta}(t) + 3\sigma_{\alpha,\delta}(t)$. Figure 5.4 illustrates this representation. However, it is necessary to translate the functional uncertainty into that of orbital elements. The Unscented Transform (UT) could be applied here, but we advocate a simpler approach: supposing the covariance matrix of orbital elements to be diagonal, only three representative curves are necessary, as opposed to the simplest UT scheme (simplex) that demands $d_\Theta + 1 = 7$ sigma-points, where d_Θ is the dimension of the orbital elements vector. In order to convert to orbital elements we apply the core IOD algorithm to each of the three curves, from which we compute the mean and a diagonal covariance matrix for the orbital elements. Let $\mathcal{T} : \mathbb{R}^2 \mapsto \mathbb{R}^6$ be the IOD map such that $\boldsymbol{\Theta}_0 := \mathcal{T}(\gamma)$, indicating the set of operations on three points (two extremal, one middle point) of the curve γ to generate the orbital elements $\boldsymbol{\Theta}_0$. Then the mean and covariance matrix for the orbital elements are taken as

$$\bar{\boldsymbol{\Theta}}_0 = \mathcal{T}(\mu_{\alpha,\delta}), \tag{5.50}$$

$$\boldsymbol{\Sigma}_\Theta \approx \frac{\operatorname{diag}\left\{\Delta\boldsymbol{\Theta}_{-3\sigma} \circ \Delta\boldsymbol{\Theta}_{-3\sigma} + \Delta\boldsymbol{\Theta}_{+3\sigma} \circ \Delta\boldsymbol{\Theta}_{+3\sigma}\right\}}{18}, \tag{5.51}$$

where $\circ$ is the Hadamard product and

$$\Delta\boldsymbol{\Theta}_{-3\sigma} = \mathcal{T}(\mu_{\alpha,\delta} - 3\sigma_{\alpha,\delta}) - \bar{\boldsymbol{\Theta}}_0, \tag{5.52}$$

$$\Delta\boldsymbol{\Theta}_{+3\sigma} = \mathcal{T}(\mu_{\alpha,\delta} + 3\sigma_{\alpha,\delta}) - \bar{\boldsymbol{\Theta}}_0. \tag{5.53}$$

For single-measurement objects such as those mentioned in Section 5.3.1, one must make assumptions on statistical deviations that cannot be determined via parametric curves. Typical electro-optical telescopes provide topocentric angles, (α_t, δ_t), with measurement noise specified with $\sigma_{\alpha,m} = \sigma_{\delta,m} = 0.001^\circ$. Thus, mean and extremal estimates of slant-range and geocentric angles, (ρ, α, δ), can be computed using Roy's procedure for a presumed geostationary radius, based on three cases: (α_t, δ_t), $(\alpha_t - 3\sigma_{\alpha,m}, \delta_t - 3\sigma_{\delta,m})$, and $(\alpha_t + 3\sigma_{\alpha,m}, \delta_t + 3\sigma_{\delta,m})$. These cases produce the sets of parameters respectively: $(\bar{\rho}, \bar{\alpha}, \bar{\delta})$, $(\bar{\rho}, \bar{\alpha} - 3\sigma_{\alpha-}, \bar{\delta} - 3\sigma_{\delta-})$, $(\bar{\rho}, \bar{\alpha} + 3\sigma_{\alpha+}, \bar{\delta} + 3\sigma_{\delta+})$. Let $\mathcal{T}' : \mathbb{R}^3 \mapsto \mathbb{R}^4$ be another IOD map that implements the routine established in Section 5.3.1 such that $(\imath, \Omega, \omega, M)^T \coloneqq \mathcal{T}'(\rho, \alpha, \delta)$ for single-measurement objects. For each of the cases, and further supposing[4] $\sigma_{a-} = \sigma_{a+} = 10^{-8}$ km and $\sigma_{e-} = \sigma_{e+} = 10^{-12}$, one obtains:

$$\bar{\boldsymbol{\Theta}}_0 = (\bar{a}, \bar{e}, \bar{\imath}, \bar{\Omega}, \bar{\omega}, \bar{M})^T, \tag{5.54}$$

$$\Delta\boldsymbol{\Theta}_{-3\sigma} = 3(\sigma_{a-}, \sigma_{e-}, \sigma_{\imath-}, \sigma_{\Omega-}, \sigma_{\omega-}, \sigma_{M-})^T, \tag{5.55}$$

$$\Delta\boldsymbol{\Theta}_{+3\sigma} = 3(\sigma_{a+}, \sigma_{e+}, \sigma_{\imath+}, \sigma_{\Omega+}, \sigma_{\omega+}, \sigma_{M+})^T, \tag{5.56}$$

where $\bar{a} = a_{\text{geo}}$, $\bar{e} = 0$, and

$$(\bar{\imath}, \bar{\Omega}, \bar{\omega}, \bar{M})^T = \mathcal{T}'(\bar{\rho}, \bar{\alpha}, \bar{\delta}), \tag{5.57}$$

$$3(\sigma_{\imath-}, \sigma_{\Omega-}, \sigma_{\omega-}, \sigma_{M-})^T = +\mathcal{T}'(\bar{\rho}, \bar{\alpha}, \bar{\delta}) - \mathcal{T}'(\bar{\rho}, \bar{\alpha} - 3\sigma_{\alpha-}, \bar{\delta} - 3\sigma_{\delta-}), \tag{5.58}$$

$$3(\sigma_{\imath+}, \sigma_{\Omega+}, \sigma_{\omega+}, \sigma_{M+})^T = -\mathcal{T}'(\bar{\rho}, \bar{\alpha}, \bar{\delta}) + \mathcal{T}'(\bar{\rho}, \bar{\alpha} + 3\sigma_{\alpha+}, \bar{\delta} + 3\sigma_{\delta+}). \tag{5.59}$$

Given parameters obtained in (5.54), (5.55), and (5.56), one can directly apply (5.51).

5.5 Association of observations

The association problem is solved in a probabilistic sense, based on a special likelihood function designed to compare pairs of observation series collected in different sessions. The analysis applies to a campaign of observations comprising a finite number of sessions, from which the observed objects are processed altogether as a single batch. As previously explained, our focus is on solving associations for any such campaign, rather than trying to track each object in real time. In this setting, any sequence of measurements, usually covering no more than two minutes each, is treated as a "tentative object" for which our system determines a state vector and its corresponding covariance matrix. Evaluating the association likelihood between pairs of objects then results in a table where all tentative objects are identified both in rows and columns, and cells intersecting any row and column contain a likelihood value of associating the corresponding objects.

The likelihood function is modeled in the ethos of the Probabilistic Data Association (PDA) [6], but with several peculiarities to be presented shortly. One major particularity is that the sta-

[4] This value assumes that uncertainty in the semi-major axis is solely due to the IOD algorithm, i.e., we do not consider whether or not the orbit is indeed in the GEO belt.

tistical comparison operates on a mixed space composed of orbital elements and measurements, i.e., the space of $\mathbf{y} \in \mathbb{R}^5$, $\mathbf{y} = (a, e, \imath, \alpha, \delta)^T$. At a first glance, operating on a 5-dimensional space might not seem advantageous because of the well-known curse of dimensionality. Nonetheless, this is exactly why the method evidences merit: sparsity of statistical events in such space enables a better matching of actually correlated objects while making erroneous associations much less probable. Clearly, had we merely considered the observational space much more association ambiguity would have been manifest, as a crude proximity of observations in the same field of view would greatly increase the number of association hypotheses. It is also important to mention that the state variables $\mathbf{y} = (a, e, \imath, \alpha, \delta)^T$ were carefully chosen to work around ambiguity in the orbital phase parameters, namely right ascension of the ascending node Ω, argument of perigee ω and mean anomaly M. Indeed (Ω, ω, M) define the orbital phase well, but their inherent trigonometric ambiguity implies that infinitely many combinations of (Ω, ω, M) could represent each pair (α, δ), making statistical comparisons unnecessarily dubious. This is especially true for nearly circular orbits that produce a rather uncertain value of ω.

Another special setting of the likelihood takes into account a two-way connection for each pair of objects, i.e., it entails that one object state must be propagated forward in time to be compared to that of another object, which in turn is propagated backward to "confirm" the association. This becomes logical as one should perceive the unnormalized likelihood table to be necessarily symmetric since the association event should be naturally commutative ($\Pr(A \to B) \equiv \Pr(B \to A)$). The trivial way to ensure a symmetric unnormalized table while considering prediction uncertainties for both objects in a pair, is by having both directions forward and backward in time, considered in the numerical evaluation. This scheme also plays in favor of robustness, as some sort of smoothing feature is implicitly incorporated in the likelihood function due to the backward confirmation.

5.5.1 Likelihood function

Herein, for the sake of brevity in the formulation, though without loss of generality, we drop conditioning on the prior information ($a \in \mathcal{A}_{\text{geo}}$) as identified in Section 5.4. Suppose that there are N tentative objects observed in a campaign. We write the random vector

$$\mathbf{\Theta}_i(t_i) = (a_i, e_i, \imath_i, \Omega_i, \omega_i, M_i)_{t_i}, \quad i \in \{1, 2, \ldots, N\},$$

to represent orbital elements of a tentative object i, originally observed at the reference epoch t_i, the time of the first measurement of the corresponding sequence. We model the pairwise association event as the correspondence map

$$\mathfrak{a}(i) : \{1, 2, \ldots, N\} \mapsto \{0, 1, 2, \ldots, N\} \setminus \{i\}, \tag{5.60}$$

i.e., for each tentative object i identified in $\mathcal{I} = \{1, \ldots, N\}$ there may be a correspondence $\mathfrak{a}(i)$ to another tentative object indexed in $\mathfrak{A}(\mathcal{I}) = \{0, 1, \ldots, N\} \setminus \{i\}$, excluding the original object itself, where $\mathfrak{a}(i) = 0$ establishes a null hypothesis meaning that object i may correspond to none of the remaining objects. We also define the prior probability that a tentative object can be observed more than once P_O^i. False alarms (clutter) are expected due to objects that occupy the surveillance region but are not members of the set of objects that are eligible to be associated. The number of false alarms is Poisson-distributed with mean $\lambda_c V$, and their

positions are assumed to be uniformly distributed in the surveillance region with volume V.

For evaluating the association likelihood, the ith object state $\boldsymbol{\Theta}_i$ must be statistically compared to the $\mathfrak{a}(i)$th object state $\boldsymbol{\Theta}_{\mathfrak{a}(i)}$ at appropriate epochs. One propagates the ith object state forward in time as $\boldsymbol{\Theta}_i(t_i) \to \boldsymbol{\Theta}_i(t_{\mathfrak{a}(i)})$, where $t_{\mathfrak{a}(i)}$ is the reference epoch for a possibly associated object $\mathfrak{a}(i)$ in order to perform the comparison $\boldsymbol{\Theta}_i(t_{\mathfrak{a}(i)}) \sim \boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_{\mathfrak{a}(i)})$. Similarly, one propagates the $\mathfrak{a}(i)$th object state backward in time as $\boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_{\mathfrak{a}(i)}) \to \boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_i)$ to perform the comparison $\boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_i) \sim \boldsymbol{\Theta}_i(t_i)$. The statistical comparisons are evaluated in the space of $\mathbf{y} \in \mathbb{R}^5, \mathbf{y} = (a, e, \dot{\imath}, \alpha, \delta)^T$, i.e., the likelihood function of each single association event can be written as

$$p(\mathfrak{a}(i)|\boldsymbol{y}_{1:N}) \triangleq \begin{cases} \frac{P_O^i}{2}\left(\mathcal{L}_f(\boldsymbol{y}_i, \boldsymbol{y}_{\mathfrak{a}(i)}) + \mathcal{L}_b(\boldsymbol{y}_{\mathfrak{a}(i)}, \boldsymbol{y}_i)\right), & \mathfrak{a}(i) \in \mathfrak{A}(\mathcal{I}) \setminus \{0\}; \\ \lambda_c(1 - P_O^i), & \mathfrak{a}(i) = 0; \end{cases} \tag{5.61}$$

where the forward and backward terms are defined respectively as

$$\mathcal{L}_f(\mathbf{y}_i, \mathbf{y}_{\mathfrak{a}(i)}) \triangleq \mathbb{E}_{\boldsymbol{\Theta}_i(t_{\mathfrak{a}(i)})}\left[p(\mathbf{y}_{\mathfrak{a}(i)}|\boldsymbol{\Theta}_i(t_{\mathfrak{a}(i)}))\right], \tag{5.62}$$

$$\mathcal{L}_b(\mathbf{y}_{\mathfrak{a}(i)}, \mathbf{y}_i) \triangleq \mathbb{E}_{\boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_i)}\left[p(\mathbf{y}_i \quad |\boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_i))\right]. \tag{5.63}$$

In the discussion above, we have assumed $t_i < t_{\mathfrak{a}(i)}$ to designate forward and backward terms, as if tentative objects are processed in ascending order of (reference) time. If another order is used the same equations apply, but the qualifiers "forward" and "backward" might be improper. The expectations in (5.62) and (5.63) are given by

$$\mathbb{E}_{\boldsymbol{\Theta}_i(t_{\mathfrak{a}(i)})}\left[p(\mathbf{y}_{\mathfrak{a}(i)}|\boldsymbol{\Theta}_i(t_{\mathfrak{a}(i)}))\right] = \int_{\mathbb{R}^6} p(\mathbf{y}_{\mathfrak{a}(i)}|\boldsymbol{\Theta}_i(t_{\mathfrak{a}(i)}))p(\boldsymbol{\Theta}_i(t_{\mathfrak{a}(i)}))d\boldsymbol{\Theta}_i(t_{\mathfrak{a}(i)}), \tag{5.64}$$

$$\mathbb{E}_{\boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_i)}\left[p(\mathbf{y}_i \quad |\boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_i))\right] = \int_{\mathbb{R}^6} p(\mathbf{y}_i \quad |\boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_i))p(\boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_i))d\boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_i), \tag{5.65}$$

where

$$\begin{aligned} p(\mathbf{y}_{\mathfrak{a}(i)}|\boldsymbol{\Theta}_i(t_{\mathfrak{a}(i)})) &= \mathcal{N}(\mathbf{y}_{\mathfrak{a}(i)}; \mathbf{h}[\bar{\boldsymbol{\Theta}}_i(t_{\mathfrak{a}(i)})], \boldsymbol{\Sigma}_{y_{\mathfrak{a}(i)}}), \\ p(\mathbf{y}_i \quad |\boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_i)) &= \mathcal{N}(\mathbf{y}_i \quad ; \mathbf{h}[\bar{\boldsymbol{\Theta}}_{\mathfrak{a}(i)}(t_i)], \boldsymbol{\Sigma}_{y_i} \quad). \end{aligned} \tag{5.66}$$

Function $\mathbf{h} : \mathbb{R}^6 \to \mathbb{R}^5$ is a pseudo-observation function that translates vectors from the orbital elements space to the mixed space according to the model presented in Section 5.2, for $\nu \approx M$:

$$\mathbf{h}[\boldsymbol{\Theta}] = \begin{pmatrix} a \\ e \\ \dot{\imath} \\ \Omega + \arctan\left[\tan(\omega + M) \cdot \cos\dot{\imath}\right] \\ \quad + \arcsin\left[\sin(\omega + M) \cdot \sin\dot{\imath}\right] \end{pmatrix}. \tag{5.67}$$

The corresponding pseudo-observation equation is defined by

$$\mathbf{y}_i = \mathbf{h}[\bar{\boldsymbol{\Theta}}_i] + \boldsymbol{\Sigma}_{y_i}^{1/2}\mathbf{v}_y, \quad \mathbf{v}_y \sim \mathcal{N}(0_5; \mathbb{I}_5), \tag{5.68}$$

with covariance matrix

$$\begin{aligned}\boldsymbol{\Sigma}_{y_i} &= \mathbf{H}(\bar{\boldsymbol{\Theta}}_i)\boldsymbol{\Sigma}_{\Theta_i}\mathbf{H}(\bar{\boldsymbol{\Theta}}_i)^T, \\ \mathbf{H}(\bar{\boldsymbol{\Theta}}_i) &= \mathbf{J}_{\boldsymbol{\Theta}}[\mathbf{h}](\bar{\boldsymbol{\Theta}}_i), \\ \mathbf{J}_{\boldsymbol{\Theta}}[\mathbf{h}] &= [\mathrm{J}_{kl}] = \left[\frac{\partial h_k}{\partial \Theta_l}\right],\ k = 1,\ldots,5;\ l = 1,\ldots,6.\end{aligned} \tag{5.69}$$

5.5.2 Forward and backward predictions

Propagation forward in time, $\boldsymbol{\Theta}_i(t_i) \to \boldsymbol{\Theta}_i(t_{\mathfrak{a}(i)})$, and backward, $\boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_{\mathfrak{a}(i)}) \to \boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_i)$, result in predicted probability densities $p(\boldsymbol{\Theta}_i(t_{\mathfrak{a}(i)}))$ and $p(\boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_i))$ respectively, which are obtained via Chapman-Kolmogorov equation:

$$p(\boldsymbol{\Theta}_i(t_{\mathfrak{a}(i)})) = \int_{\mathbb{R}^6} p(\boldsymbol{\Theta}_i(t_{\mathfrak{a}(i)})|\boldsymbol{\Theta}_i(t_i))p(\boldsymbol{\Theta}_i(t_i))d\boldsymbol{\Theta}_i(t_i), \tag{5.70}$$

$$p(\boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_i)) = \int_{\mathbb{R}^6} p(\boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_i)|\boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_{\mathfrak{a}(i)}))p(\boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_{\mathfrak{a}(i)}))d\boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_{\mathfrak{a}(i)}). \tag{5.71}$$

The probability densities $p(\boldsymbol{\Theta}_i(t_i))$ and $p(\boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_{\mathfrak{a}(i)}))$ are defined as

$$p(\boldsymbol{\Theta}_j(t_j)) = \mathcal{N}(\boldsymbol{\Theta}_j(t_j); \bar{\boldsymbol{\Theta}}_j(t_j), \boldsymbol{\Sigma}_{\Theta_j}), \quad j = i, \mathfrak{a}(i), \tag{5.72}$$

and the moments of orbital elements $\bar{\boldsymbol{\Theta}}_j(t_j)$ and $\boldsymbol{\Sigma}_{\Theta_j}$ are determined according to the method presented in Section 5.4, as per equations (5.50) and (5.51). The densities $p(\boldsymbol{\Theta}_i(t_{\mathfrak{a}(i)})|\boldsymbol{\Theta}_i(t_i))$ and $p(\boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_i)|\boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_{\mathfrak{a}(i)}))$ describe the Markov processes with transition kernels

$$p(\boldsymbol{\Theta}_i(t_{\mathfrak{a}(i)})|\boldsymbol{\Theta}_i(t_i)) = \mathcal{N}(\boldsymbol{\Theta}_i(t_{\mathfrak{a}(i)}); \mathbf{F}_f\bar{\boldsymbol{\Theta}}_i(t_i) + \mathbf{G}_f, \mathbf{Q}_f), \tag{5.73}$$

$$p(\boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_i)|\boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_{\mathfrak{a}(i)})) = \mathcal{N}(\boldsymbol{\Theta}_{\mathfrak{a}(i)}(t_i); \mathbf{F}_b\bar{\boldsymbol{\Theta}}_{\mathfrak{a}(i)}(t_{\mathfrak{a}(i)}) + \mathbf{G}_b, \mathbf{Q}_b), \tag{5.74}$$

where linearized transition matrices $\mathbf{F}_f$ and $\mathbf{F}_b$ are defined for the forward and backward processes respectively, and both processes have the same offset vector $\mathbf{G} = \mathbf{G}_f = \mathbf{G}_b$ and the same process noise covariance $\mathbf{Q} = \mathbf{Q}_f = \mathbf{Q}_b$. As discussed previously, the transition models are intended to delineate an unperturbed motion in which dominant non-secular and short-periodic effects are captured as uncertainty, i.e., the process noise partially compensates for modeling simplification and errors. The Markov processes are described by the discrete-time linear state-space models

$$\boldsymbol{\Theta}(t_{k+1}) = \mathbf{F}_{f,k}\boldsymbol{\Theta}(t_k) + \mathbf{G}_{f,k} + \mathbf{Q}_k^{1/2}\mathbf{w}, \tag{5.75}$$

$$\boldsymbol{\Theta}(t_{k-1}) = \mathbf{F}_{b,k}\boldsymbol{\Theta}(t_k) + \mathbf{G}_{b,k} + \mathbf{Q}_k^{1/2}\mathbf{w}, \tag{5.76}$$

$$\mathbf{w} \sim \mathcal{N}(0_6, \mathbb{I}_6), \tag{5.77}$$

where

$$\mathbf{F}_{f,k} = \begin{pmatrix} 1 & 0 & \ldots & 0 \\ 0 & 1 & & \\ \vdots & & \ddots & \\ +f_{61}T & & & 1 \end{pmatrix},\ \mathbf{G}_{f,k} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ g_6T \end{pmatrix}, \tag{5.78}$$

$$\mathbf{F}_{b,k} = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & & \\ \vdots & & \ddots & \\ -f_{61}T & & & 1 \end{pmatrix}, \mathbf{G}_{b,k} = \mathbf{G}_{f,k}, \tag{5.79}$$

for $f_{61} = -\frac{3n_k}{2a_k}$, $g_6 = \frac{5}{2}n_k$ and $T = \Delta t$. The process noise covariance matrix is given by

$$\mathbf{Q} = \begin{pmatrix} \sigma_{\delta a}^2 T & 0 & & \dots & 0 & \sigma_{\delta a}^2 f_{61}\frac{T^2}{2} \\ 0 & \sigma_{\delta e}^2 T & & & & 0 \\ \vdots & & \sigma_{\delta i}^2 T & & & \\ & & & \sigma_{\delta\Omega}^2 T & & \vdots \\ 0 & & & & \sigma_{\delta\omega}^2 T & 0 \\ \sigma_{\delta a}^2 f_{61}\frac{T^2}{2} & 0 & \dots & & 0 & \sigma_{\delta M}^2 T + \sigma_{\delta a}^2 f_{61}^2 \frac{T^3}{3} \end{pmatrix}. \tag{5.80}$$

According to [200] (pages 187-191), for $e_k \approx 0$ and small inclination ($i_k \leq 15°$), the following approximations are valid

$$\sigma_{\delta a} \approx \frac{3}{2}\bar{\psi}_2\, a_k(1 - 3\cos^2 i_k)e_k\,(\cos(M_k + \sigma_{\delta M}) - \cos M_k)\,, \tag{5.81}$$

$$\sigma_{\delta e} \approx \frac{3}{4}\bar{\psi}_2\,(1 - e_k^2)(1 - 3\cos^2 i_k)\,(\sin M_k + 3e_k \sin 2M_k)\,, \tag{5.82}$$

$$\sigma_{\delta i} \approx \frac{3}{2}\bar{\psi}_2\,(1 - e_k^2)^{-1/2} \sin i_k \cos i_k \sin(2\omega_k + 2M_k), \tag{5.83}$$

$$\sigma_{\delta\Omega} \approx \frac{3}{2}\bar{\psi}_2\,(1 - e_k^2)^{-1/2}\,(-1 + \cos(2\omega_k + 2M_k))\cos i_k, \tag{5.84}$$

$$\sigma_{\delta\omega} \approx \frac{3}{4}\bar{\psi}_2\,(5\cos^2 i_k - 1), \tag{5.85}$$

$$\sigma_{\delta M} \approx \frac{3}{2}\bar{\psi}_2\left(1 - \frac{3}{2}\sin^2 i_k\,(1 - \cos(2\omega_k + 2M_k))\right), \tag{5.86}$$

for parameters in canonical units, where $\bar{\psi}_2 = n_k r_E^2 J_2/a_k^2$, r_E is the Earth's equatorial radius, and J_2 is the second-order zero-degree zonal harmonic of the Earth's gravitational field. The discrete-time model expressed by equations (5.75)–(5.86) is obtained by the method presented in Appendix A. Equations (5.81)–(5.86) refer to the evolving uncertainty that models the averaged orbit perturbations due to the major components of Earth's non-spherical terms, accounting for the time-varying effects of Earth's oblateness. In this chapter, the luni-solar attraction effects are not considered in (5.81)–(5.86). For reference, let us consider a stabilized orbit, disconsidering an offset of about $+2\,\text{km}$ in the semi-major axis due to J_2, which is slighly compensated by an offset of about $-1\,\text{km}$ due to the luni-solar attraction [120]. In this condition, the averaged magnitude of Earth's non-spherical perturbations in the orbital elements, for a geostationary object, are of order[5] $\delta a \sim 10^{-8}\,\text{m}$, $\delta e \sim 10^{-4}$, $\delta i \sim 10^{-6}\,\text{rad}$, $\delta\Omega \sim 10^{-4}\,\text{rad}$, $\delta\omega \sim 10^{-4}\,\text{rad}$, $\delta M \sim 10^{-4}\,\text{rad}$ per day, and the averaged magnitude of luni-solar attraction perturbations are of order[6] $\delta a \sim 10^{-12}\,\text{m}$, $\delta e \sim 10^{-5}$, $\delta i \sim 10^{-5}\,\text{rad}$, $\delta\Omega \sim 10^{-5}\,\text{rad}$, $\delta\omega \sim$

[5]The perturbation due to Earth's non-spherical gravitational field in the semi-major axis produces an offset of about $+2\,\text{km}$ due to J_2, a negligible secular term per day, and a libration of amplitude $\delta a \sim 20\,\text{m}$ induced by tesseral terms (J_{22}).

[6]The perturbation due to the luni-solar attraction in the semi-major axis produces an average offset of about $-1\,\text{km}$, a negligible secular term per day, and a short-periodic libration of amplitude $\delta a \sim 1\,\text{km}$ depending on the dihedral angles between the sun/moon and the satellite and on the lunar phase.

10^{-5} rad, $\delta M \sim 10^{-5}$ rad per day [120].

Notice that, under the Gaussianity assumptions, it is possible to demonstrate that the forward (5.62) and backward (5.63) components of the association likelihood can be evaluated as

$$\mathcal{L}_{f/b}(\mathbf{y}_u, \mathbf{y}_v) = \mathcal{N}(\mathbf{y}_u; \bar{\boldsymbol{y}}_v, \mathbf{P}_{y_u y_v}), \tag{5.87}$$

where, for $u, v \in \{i, \mathfrak{a}(\mathfrak{i})\}$ and $\mathbf{F} \in \{\mathbf{F}_f, \mathbf{F}_b\}$:

$$\bar{\boldsymbol{\Theta}}_{uv} \coloneqq \bar{\boldsymbol{\Theta}}_u(t_v) = \mathbf{F}\bar{\boldsymbol{\Theta}}_u(t_u) + \mathbf{G}, \tag{5.88}$$

$$\bar{\boldsymbol{y}}_u = \mathbf{h}[\bar{\boldsymbol{\Theta}}_{uv}], \tag{5.89}$$

$$\mathbf{P}_{y_u y_v} = \mathbf{H}(\bar{\boldsymbol{\Theta}}_{uv})(\mathbf{F}\boldsymbol{\Sigma}_{\Theta_u}\mathbf{F}^T + \mathbf{Q})\mathbf{H}(\bar{\boldsymbol{\Theta}}_{uv})^T + \boldsymbol{\Sigma}_{y_v}. \tag{5.90}$$

5.5.3 Outline of the association algorithm

The association algorithm is now briefly outlined.

Input data

Campaign of observations providing multiple-session batches.

Step 0

The data is structured in clusters, where each cluster collects objects from the same observation session. This is done to divide the problem and promote computational efficiency since only tentative objects from different clusters are compared.

Step 1

The IOD algorithm is applied to parametric curves of every single tentative object i (Sections 5.3 and 5.4), generating a database for all objects containing:

- orbital elements of the mean parametric curve $\bar{\boldsymbol{\Theta}}_i$,
- covariance matrix representing uncertainty of orbital elements $\boldsymbol{\Sigma}_{\Theta_i}$.

Step 2

For each tentative object, orbital parameters are used to propagate forward its orbit corresponding to the mean curve and its uncertainty (Section 5.5.2). Orbits are propagated to match the epoch of first measurement for each tentative object from different clusters.

Step 3

For each tentative object, the orbital parameters are used to propagate backward the orbit corresponding to the mean curve and its uncertainty (Section 5.5.2).

STEP 4

Gating is used to identify pairs of tentative objects that give rise to a significant likelihood of being associated. To make the association algorithm computationally efficient, the likelihood function is only evaluated if a propagated object u falls within the validation (elliptical) region $\mathcal{V}_v(\vartheta, u)$ around object v, and propagated object v falls in validation region $\mathcal{V}_u(\vartheta, v)$ of object u, where

$$\mathcal{V}_v(\vartheta, u) = \left\{(\mathbf{y}_v - \mathbf{h}[\bar{\mathbf{\Theta}}_{uv}])\mathbf{P}^{-1}_{y_u y_v}(\mathbf{y}_v - \mathbf{h}[\bar{\mathbf{\Theta}}_{uv}]) \leq \vartheta\right\}, \tag{5.91}$$

and ϑ is a gate threshold corresponding to a certain level of confidence $1-\mathscr{c} = 99.73\%$, for $\kappa = 5$ degrees of freedom (dimension of $\mathbf{y}$), according to the chi-square inverse cumulative distribution function $\vartheta \coloneqq \chi^{-2}_{\text{cdf}}(1-\mathscr{c}, \kappa)$.

STEP 5

The pairwise association likelihood function is evaluated (Section 5.5.1) to produce an unnormalized likelihood table, whose rows and columns are identified with all tentative objects. Each cell of the table intersecting a row and a column contains the likelihood function value of associating the corresponding tentative objects.

STEP 6

The likelihood table is normalized[7] per tentative object (each row), which corresponds to assume that all possibilities have been considered: either the object is not associated to any of the remaining objects, or may have been associated to each of them. Normalized values of likelihood are then tested against a "likelihood threshold" to determine if each pairwise association is either true or false.

5.6 RESULTS FOR SIMULATED OBSERVATIONS

We tested the proposed framework for a campaign of 7 nights, each containing 2 measurement sessions, with 354 observation sequences collected from 59 objects, of which only 34 have been observed more than once. The data has been provided by Airbus Safran Launchers, and is based on simulated data for actual objects using two-line elements (TLEs) propagated by the *Simplified General Perturbations 4* (SGP4) model [93]. The measurements in azimuth and elevation (and topocentric right ascension and declination), incorporating noise and missed detections, have been generated as from a single telescope using a function validated and used in the Airbus Safran Launchers Space Surveillance System. The prior probability of an object being observed more than once was set as $P_O = 1.5\%$ and the clutter density rate was estimated to be $\lambda_c \approx 2 \cdot 10^{11}$ (false alarms/unit of V), in a surveillance hypervolume V defined in the space of $\mathbf{y} = (a, e, \imath, \alpha, \delta)^T$. The tentative objects use a non-standard identification scheme, in the format **N{nn}-M{mm}-O{ooo}** where "N" stands for a night identified by a two-digit number {nn}, "M" stands for measurement session identified by number {mm}, and "O" stands for object identified by {ooo}. This identification has no correspondence to those from available

[7] Note that once the likelihood table is normalized, including the null hypothesis (no association), it becomes asymmetric.

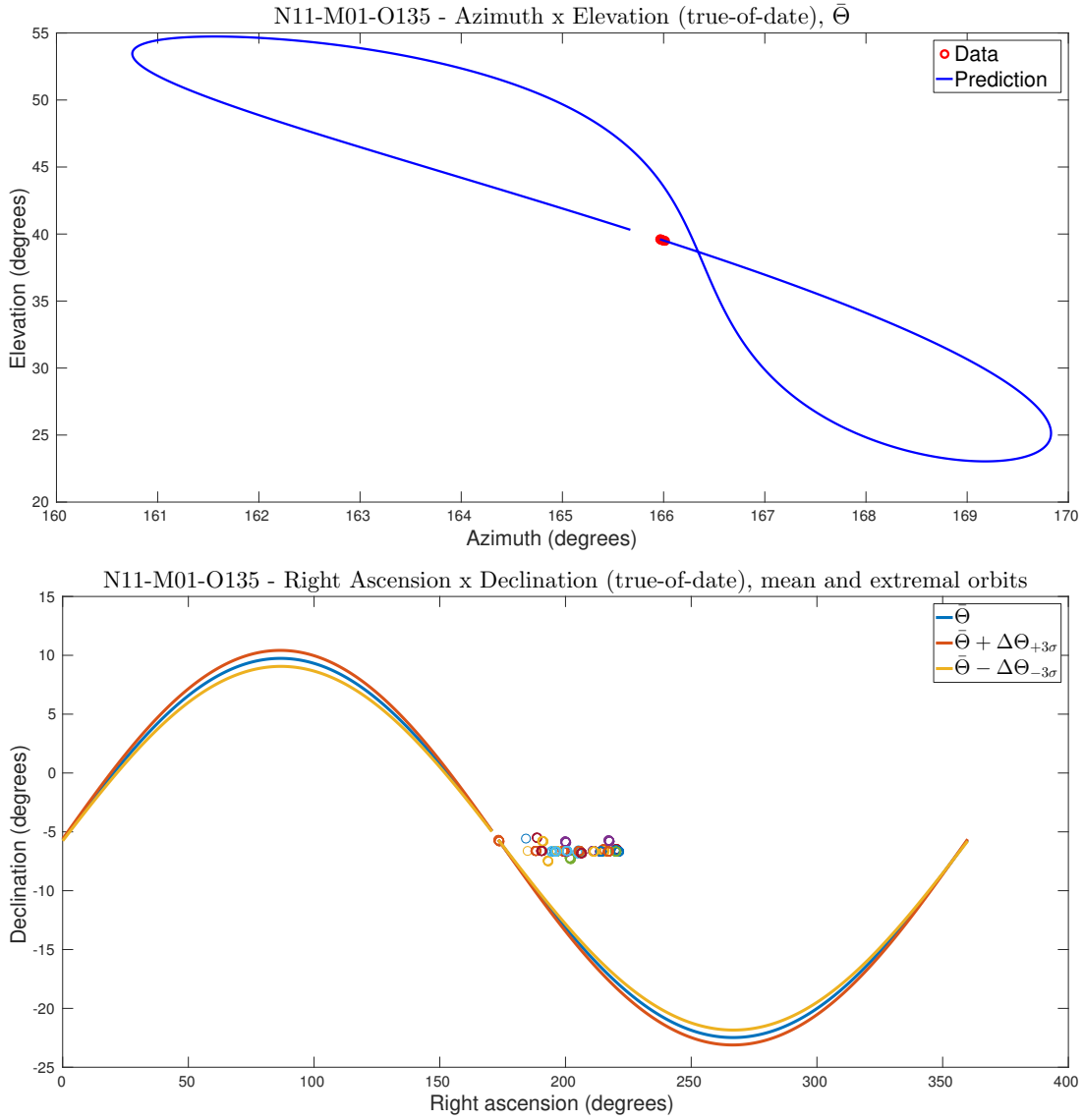


Figure 5.5: Example of orbits for parametric curves

catalogs. We remark that each observed instance of an object is given a different identification code.

The initial orbit determination of parametric curves, for all tentative objects, is successfully achieved with a single iteration, resulting in mean square errors (5.42) of order 10^{-5}. Figure 5.5 shows an example of orbits determined for the mean and extremal parametric curves of the tentative object N11-M01-O135, along with measurements of other tentative objects that are registered along one day of prediction.

Figure 5.6 exemplifies forward and backward predictions applied to tentative objects for performing the statistical comparison. Black squares mark data for a first tentative object, whose estimated orbital elements and uncertainty are used to predict trajectories of a mean parametric curve (red line) and its extremal curves (yellow and purple lines). Data for a second object is marked by blue circles. Predicted positions, along parametric curves of the first object, are identified by x's at the epoch when the second object was observed. Parametric curves are

Assoc.	Object 1	Object 2	Likelihood
1	N20-M02-O343	N24-M02-O347	100.0000%
2	N11-M02-O334	N21-M01-O343	100.0000%
3	N19-M01-O500	N19-M02-O501	99.9998%
4	N24-M01-O462	N24-M02-O463	99.9941%
5	N24-M01-O548	N24-M02-O549	99.9722%

Table 5.2: Top-five pairwise associations in terms of likelihood values.

depicted over four hours around the predictions to illustrate their behavior. In Figure 5.6 one can notice a close correspondence of the information contained in stochastic predictions with the likelihood of association. High likelihood values are obtained for these examples meaning that the corresponding associations are very likely. In the example of Figure 5.6a, the difference of observation epochs between the first and second objects is $\Delta t = +04.05\,\text{h}$, whereas in the example of Figure 5.6b it is $\Delta t = -70.50\,\text{h}$, i.e., backwards in time. The second example was selected to show a situation where a single measurement has been provided for the first and second objects, when effectiveness of the algorithm is kept even for orbits that are lowly observable.

In Table 5.2, we show the top-five pairwise associations[8] based on their likelihood values.

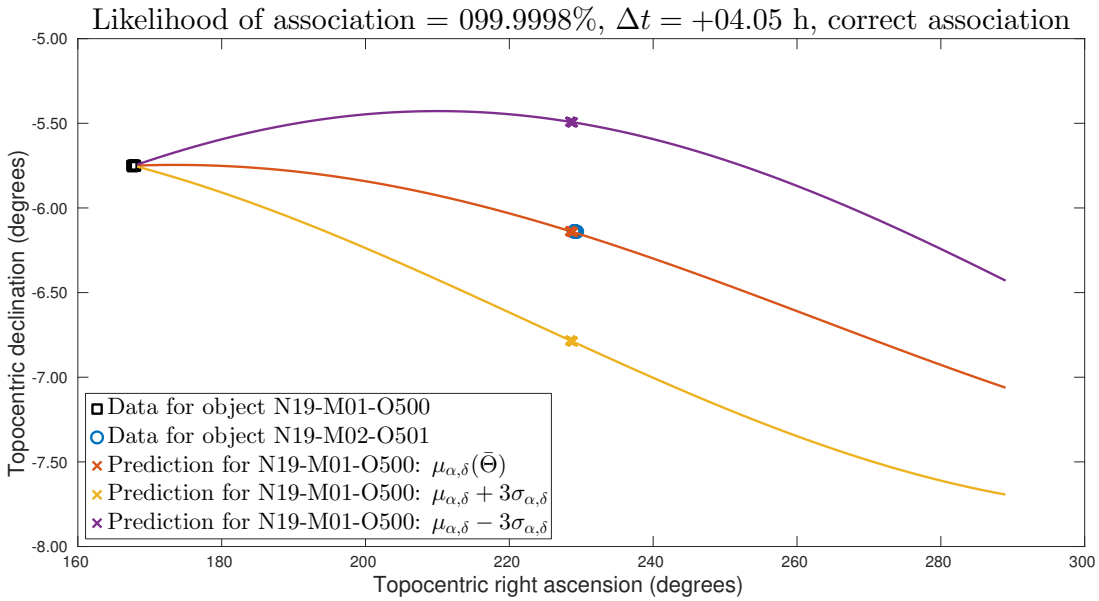
When verifying performance of threshold-based binary classifier systems the so-called "receiver operating characteristics" (ROC) curve is very useful to select possibly optimal thresholds of a decision model. The technique is very commonly applied to evaluate classifiers in machine learning applications. The association problem was modeled as a decision problem to discriminate if two observed objects are in fact the same: if the likelihood of their association is greater than a given threshold then they are declared as associated, else they are declared as non-associated. A true positive is achieved when the system declares two objects as associated and they are indeed the same object. Similarly, a true negative is achieved when the system declares two objects as non-associated and they are not the same object in fact. Defining TP as the number of true positives determined by the system, FP the number of false positives, TN the number of true negatives, and FN the number of false negatives, the true-positive rate and false-positive rate of association are defined as

$$TPR \coloneqq \frac{TP}{\#\text{ of actual positives}} = \frac{TP}{TP + FN}, \tag{5.92}$$

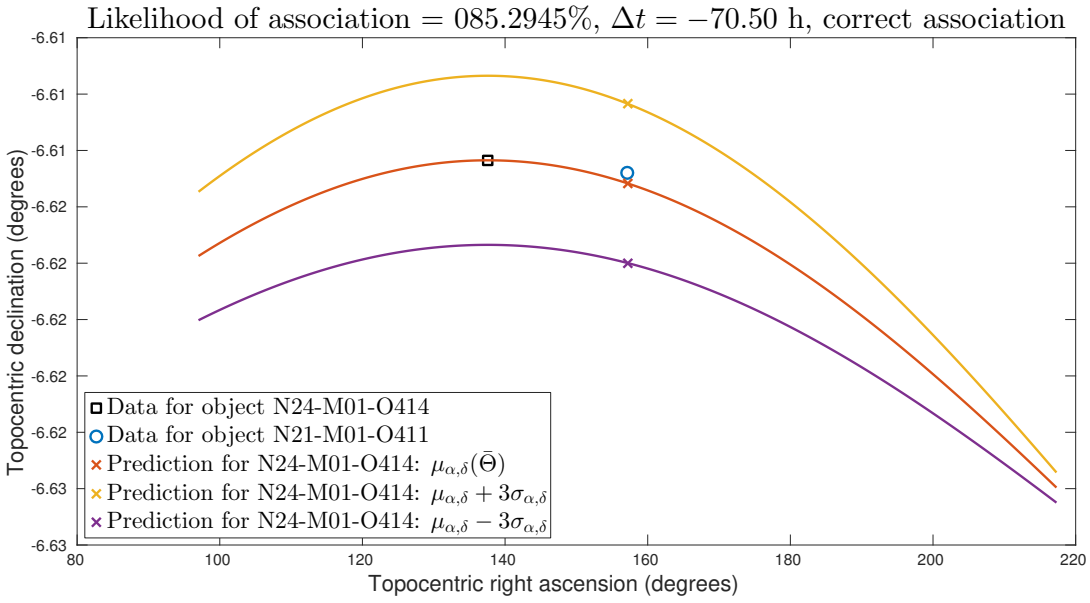
$$FPR \coloneqq \frac{FP}{\#\text{ of actual negatives}} = \frac{FP}{TN + FP}. \tag{5.93}$$

The ROC curve is generated by plotting the true-positive rate TPR (aka sensitivity, recall or probability of detection) of a classification problem against the false-positive rate FPR (aka fall-out, probability of false alarm or 1-specificity) at various threshold settings. Clearly, a perfect classifier would produce $TPR = 100\%$ and $FPR = 0\%$, and random guesses should produce $\langle TPR \rangle = \langle FPR \rangle$ for a large number of trials. Figure 5.7 shows the ROC curve of pairwise associations generated by the proposed association algorithm, for several likelihood thresholds. For the analyzed campaign, there exist 124,962 possible pairwise associations, of which 3,170 are positive. For zero threshold it is possible to notice that the association method

[8]Notice that in Table 5.2 all associations are correct, i.e., objects 1 and 2 are actually the same, but their identification codes are different because they were observed in different sessions.



(a) Forward prediction



(b) Backward prediction

Figure 5.6: Examples of forward and backward predictions

approximates very well an ideal system, which would have produced a single point at the upper left corner of the graph ($TPR = 100\%$, $FPR = 0\%$). As the threshold increases, a number of true positives is discarded at the benefit of also discarding false positives. At some threshold, around 40%, the false-positive association rate is reduced to a very negligible value while the true-positive rate is yet significant enough, despite a big deal of missed associations, to enable correct associations of most objects. This can be objectivelly perceived in Figures 5.8 and 5.9, where we grouped objects for several threshold settings, in the following categories

1. perfectly associated: objects for which all occurrences were correctly detected, no association was missed and no false association was determined;
2. almost perfectly associated: objects for which almost all occurrences were correctly detected, with some possibly missed associations but without any false association;
3. imperfectly associated: objects for which some correct associations have been detected, but both missed and false associations may have occurred;
4. objects with only one correct association.

The graph in Figure 5.8 categorizes all 59 objects, meaning that, from objects only observed once, those with no declared association take part in the "perfectly associated" category. In addition, from objects only observed once, those with mistakenly declared association (false positives) integrate the "imperfectly associated" category. From a practical standpoint, a satisfactory solution should provide one or more correct associations for most objects, possibly admitting some missing associations. In this sense, as evidenced in Figure 5.8, the solution is practically satisfactory for likelihood thresholds bigger than 40%, as 48–57 out of 59 objects ($\sim$ 81.3–96.6%) have perfect or almost perfect associations. Besides, most objects have more than one correct association. In Figure 5.9, one can verify that, when only considering objects observed more than once, the number of objects with strictly perfect associations is very small, however, the number of objects with almost perfect associations is relatively higher since, in absolute terms, it is exactly the same number in both graphs of Figures 5.8 and 5.9. In that case, 23–32 objects out of 34 ($\sim$ 67.6–94.1%) have almost perfect associations, and thus, become candidate inputs to a full-precision orbit determination method. It should be noted that, although imperfect associations are undesirable, they are still useful if only one correct association is required by the full-precision orbit determination method. This becomes clear by recalling the high probability of correct (pairwise) association the algorithm provides, as represented by the relatively high true-positive rate reported in the ROC curve.

Preliminary tests with state-of-the-art IOD methods (Gauss, Laplace, Gooding) were performed and suggest that, specifically for the problem posed in this chapter, our proposed IOD method is more convenient, enabling plausible values of association likelihoods. Unfortunately, we were not able to make standard IOD methods to work properly on the complete set of observations, which prevented us to elaborate a fair comparison in our numerical experiments. We believe this difficulty is related to the fact that standard methods, being quite general, struggle with sets of observations covering very short arcs (that demand specific assumptions). In contrast, the IOD algorithm we developed has been demonstrated to deal well with such cases.

The complete step of IOD runs, for all 354 tentative objects, takes only about 11 seconds, i.e., around 0.031 seconds per object. The likelihood evaluations take 0.0004 seconds per pair

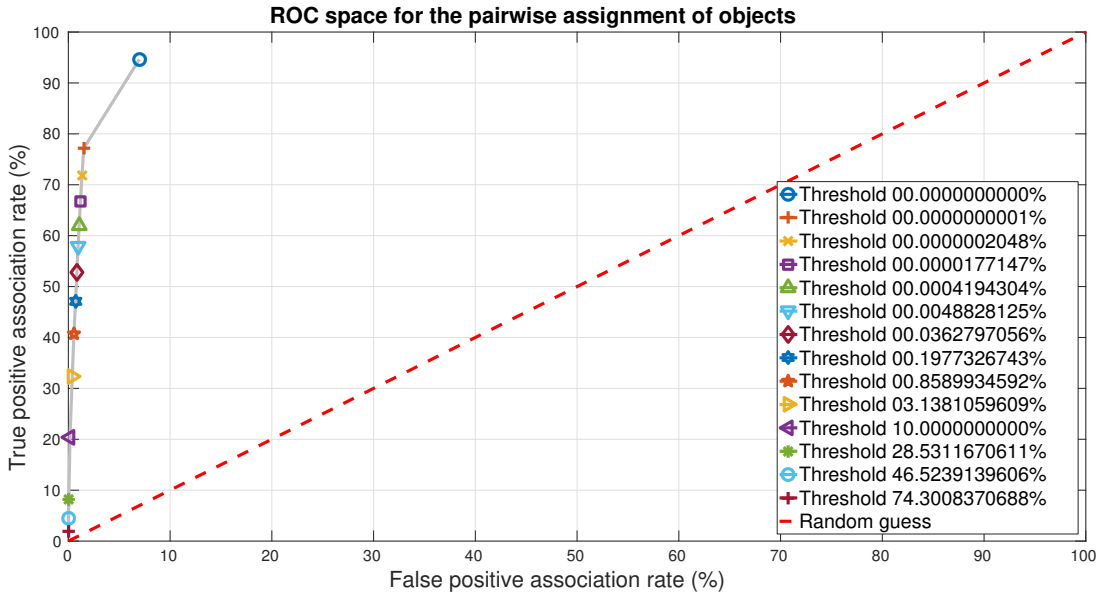


Figure 5.7: ROC curve of the association algorithm

of objects. In the worst case, by removing the gating step and evaluating every possible pair of objects, the whole process takes around 61 seconds. These computation times refer to programs written in Matlab 2016a, running on a personal computer featuring a CPU Intel Core i7-4700HQ running at 2.40 GHz (8 cores), with 2 Mb of cache memory, 8 Gb of RAM, under a Linux 64-bit operating system.

5.7 Conclusions and future work

The issue of resolving data association ambiguity and representing uncertainty accurately when the state observability is low, as identified in Chapter 0, has been addressed in this chapter. A solution to this critical issue has been successfully proposed in the form of a framework of integrated algorithms for resolving the association ambiguities and representing the uncertainty to an adequate level of accuracy. Specifically, this has been achieved for a difficult Space Surveillance scenario where the sensor is not able to provide angle rates, ranges, or range rates for nearly-geostationary or geosynchronous objects orbiting the Earth.

The solution involves a novel system to perform observation-to-observation associations for objects in the geostationary belt, with low observability due to short arc lengths covered by each observation. The proposed framework is based on a new initial orbit determination method, derived to enable a reasonable description of nearly-geosynchronous and geostationary orbits and their uncertainties, and a procedure for statistical comparison between estimated orbits in a mixed space comprising orbital elements and measurements. Any two objects observed in different epochs can then be compared to generate likelihood values that quantify their similarity. The method partially works around the computational complexity implied in standard multiple-hypothesis filtering and focuses on producing lists of associated objects before a full-precision orbit determination algorithm can be applied.

The proposed IOD algorithm has been demonstrated to provide reasonably good estimates in the context of angles-only, very short-arc orbit determination. It is also worth mentioning

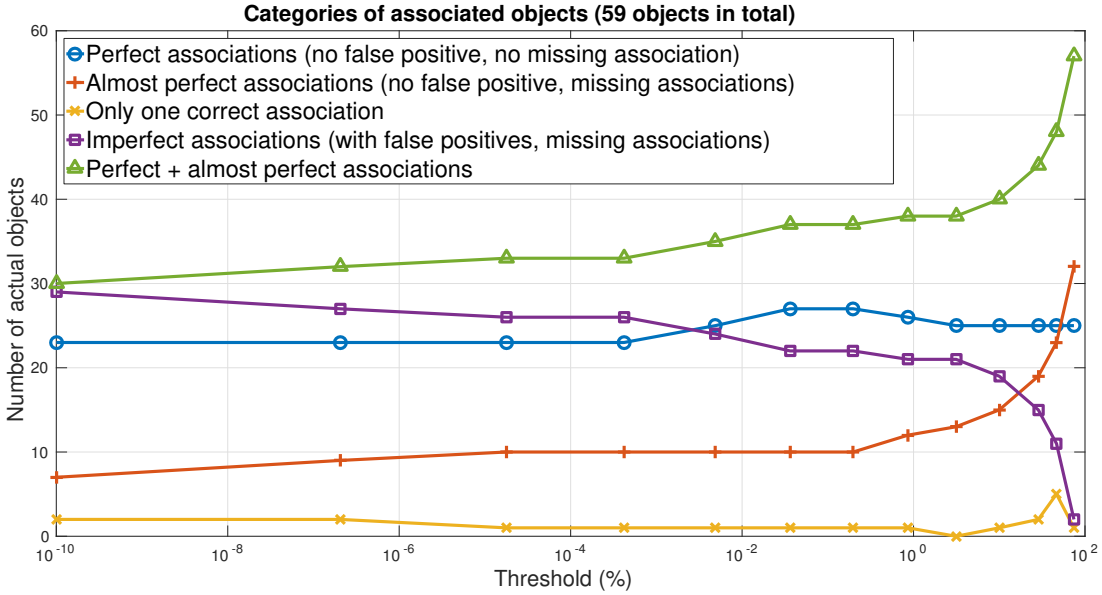


Figure 5.8: Categories of all objects

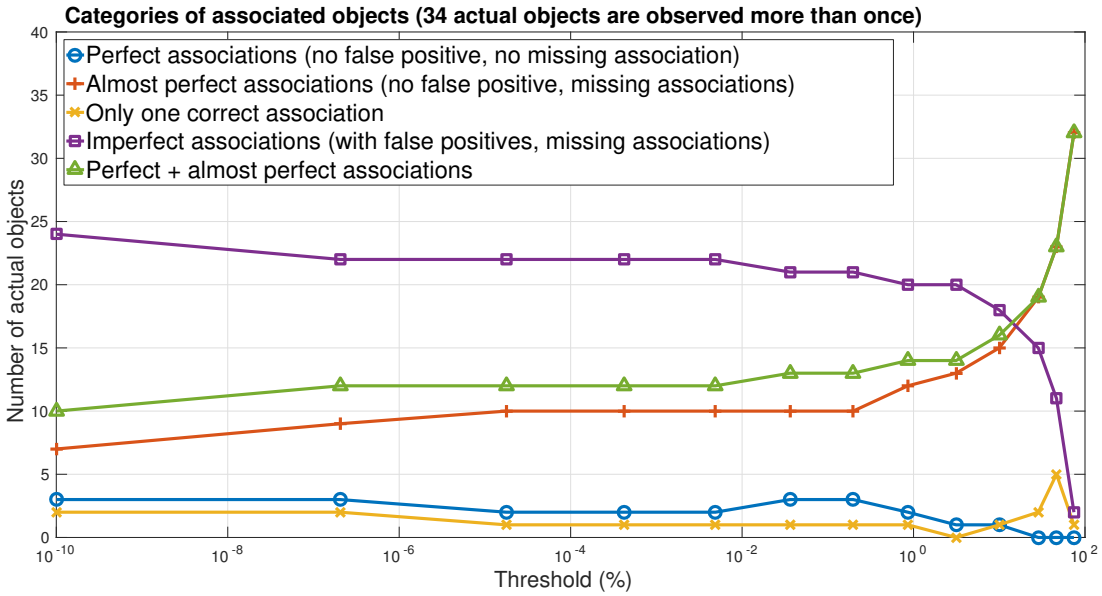


Figure 5.9: Categories of recurring objects

that the IOD algorithm could be altered to either incorporate a model of synchronous elements [184], or update slant-ranges by solving the Lambert problem.

We note that the modeled likelihood function makes a bold assumption: non-observed objects are regarded as non-existent. Although this is a common aspect in PDA formulation [6], one could benefit from negative information when objects are not observed where one would predict them to be. This could be incorporated by considering the probability of existence, for instance. While the proposed system avoids explicit tracking, one could think of the problem as directly estimating objects in the orbital elements' space, which could be implemented in terms of intensity filters (or Probability Hypothesis Density filters) [126], capturing features of the objects in the elements' space and avoiding explicit comparison between objects.

Regarding computational aspects, we remark that theoretically speaking, the problem is bounded by exponential complexity on the number of objects, in the worst case. However, in practice, the problem manifests itself significantly sparse when statistical comparisons are performed in a suitable space. When this sparsity is manifested in an augmented space, which both resolves ambiguities and inflates the sample space volume, clever strategies such as gating and pre-ordering of the set of tentative objects turn the problem sub-exponential in time. In practice, the tools we propose are not computationally expensive, taking only 1–2 minutes to completely process a campaign with 354 tentative objects. However, it could be made more scalable by preliminary matching of orbital elements that vary slowly with the orbit phase, under a pre-ordered geometric structure in the space of events. These geometric structures are called "kd-trees" and allow searches bounded by $\mathcal{O}(\log N)$ computations on average.

As a last remark, we highlight the relevance and moderate novelty of the method used for describing and estimating the initial orbit uncertainties. While functional data analysis is well established, it is rare to see it adopted in practical engineering applications, despite its almost perfect adherence to problems such as that posed in this chapter. We believe this should be explored as a promising avenue for dealing with uncertainty in orbit determination when one is generally interested in estimating an orbit as a whole curve rather than inferring its local properties.

In conclusion, we hope that the tools proposed herein offer a feasible solution for the observation-to-observation association problem in the orbit determination context. The case made suggests that judiciously chosen assumptions and algorithm settings may lead to outstanding results, at the cost, by design, of possibly detrimental effects to generality. As future work, we plan to enhance our system such that it can associate real measurements of nearly-geostationary ("GEO") objects as observed by several telescopes, and modify our algorithms to successfully process real measurements from Low Earth Orbit (LEO) objects.

References

[6] Y. Bar-Shalom, F. Daum, and J. Huang. "The Probabilistic Data Association Filter". In: *IEEE Control Systems* 29.6 (Dec. 2009), pp. 82–100.

[8] R. H. Battin. *An Introduction to the Mathematics and Methods of Astrodynamics*. AIAA education series. American Institute of Aeronautics & Astronautics, 1999.

[60] K. J. DeMars, M. K. Jah, and P. W. Schumacher. "Initial Orbit Determination using Short-Arc Angle and Angle Rate Data". In: *IEEE Transactions on Aerospace and Electronic Systems* 48.3 (July 2012), pp. 2628–2637.

[61] K. J. DeMars and M. K. Jah. "Probabilistic Initial Orbit Determination Using Gaussian Mixture Models". In: *Journal of Guidance, Control, and Dynamics* 36.5 (July 2013), pp. 1324–1335.

[69] P. R. Escobal. *Methods of Orbit Determination*. R. E. Krieger Publishing Company, 1976.

[71] D. Farnocchia et al. "Innovative methods of correlation and orbit determination for space debris". In: *Celestial Mechanics and Dynamical Astronomy* 107.1 (2010), pp. 169–185.

[74] K. Fujimoto and D. J. Scheeres. "Non-linear Bayesian orbit determination based on the generalized admissible region". In: *FUSION 2012, Proceedings of the 15th International Conference on Information Fusion*. July 2012, pp. 2043–2049.

[75] K. Fujimoto et al. "Association of Short-Arc Optical Tracks via the Direct Bayesian Admissible Region: Theory and Application". In: *6th European Conference on Space Debris*. Vol. 723. ESA Special Publication. Aug. 2013, p. 67.

[76] T. Fukushima. "Fast Procedure Solving Universal Kepler's Equation". In: *Celestial Mechanics and Dynamical Astronomy* 75.3 (Nov. 1999), pp. 201–226.

[77] S. M. Gadaleta, J. T. Horwood, and A. B. Poore. "Short arc gating in multiple hypothesis tracking for space surveillance". In: *Proc. SPIE 8385, Sensors and Systems for Space Applications V, 83850Y*. Vol. 8385. May 2012, 83850Y–83850Y-15.

[82] R. H. Gooding. "A new procedure for the solution of the classical problem of minimal orbit determination from three lines of sight". In: *Celestial Mechanics and Dynamical Astronomy* 66.4 (1996), pp. 387–423.

[85] G. F. Gronchi, L. Dimare, and A. Milani. "Orbit determination with the two-body integrals". In: *Celestial Mechanics and Dynamical Astronomy* 107.3 (July 2010), pp. 299–318.

[93] F. R. Hoots and R. L. Roehrich. *Spacetrack Report #3: Models for Propagation of the NORAD Element Sets*. Tech. rep. Colorado Springs, CO: U.S. Air Force Aerospace Defense Command, 1980.

[96] I. Hussein et al. "Probabilistic Admissible Region for Short-Arc Angles-only Observations". In: *AMOS Conference*. 2014.

[104] B. A. Jones, B.-T. Vo, and B.-N. Vo. "Generalized Labeled Multi-Bernoulli Space-Object Tracking with Joint Prediction and Update". In: *SPACE Conferences and Exposition*. American Institute of Aeronautics and Astronautics, Sept. 2016.

[120] H. Li. "Geostationary Satellites Collocation". In: SpringerLink: Bücher. Springer Berlin Heidelberg, 2014. Chap. 4: Geostationary Orbit Perturbation.

[126] R. P. S. Mahler. "Multitarget Bayes Filtering via First-Order Multitarget Moments". In: *IEEE Transactions on Aerospace and Electronic Systems* 39.4 (Oct. 2003), pp. 1152–1178.

[132] F. L. Markley. "Kepler Equation solver". In: *Celestial Mechanics and Dynamical Astronomy* 63.1 (Mar. 1995), pp. 101–111.

[134] J. M. Maruskin, D. J. Scheeres, and K. T. Alfriend. "Correlation of Optical Observations of Objects in Earth Orbit". In: *Journal of Guidance, Control, and Dynamics* 32.1 (Jan. 2009), pp. 194–209.

[138] A. Milani. "The Asteroid Identification Problem: I. Recovery of Lost Asteroids". In: *Icarus* 137.2 (1999), pp. 269–292.

[139] A. Milani and Z. Knežević. "From Astrometry to Celestial Mechanics: Orbit Determination with Very Short Arcs". In: *Celestial Mechanics and Dynamical Astronomy* 92.1 (2005), pp. 1–18.

[158] J. Ramsay and B. W. Silverman. *Functional Data Analysis*. 2nd. Springer Series in Statistics. Springer, 2005.

[170] A. E. Roy. *Orbital Motion*. Adam Hilger Limited, 1978.

[179] J. A. Siminski et al. "Short-arc tracklet association for geostationary objects". In: *Advances in Space Research* 53 (Apr. 2014), pp. 1184–1194.

[180] N. Singh et al. "Multiple Hypothesis Tracking (MHT) for Space Surveillance: Results and Simulation Studies". In: *Proceedings of the 2013 Advanced Maui Optical and Space Surveillance Technologies Conference (Wailea, Maui, Hawaii)*. Sept. 2013.

[181] W. M. Smart. "Celestial Mechanics". In: London: Longmans, Green, and Co., 1953, p. 38.

[184] E. M. Soop. *Handbook of Geostationary Orbits*. Space Technology Library. Springer Netherlands, 1994.

[186] J. Stauch et al. "Mutual Application of Joint Probabilistic Data Association, Filtering, and Smoothing Techniques for Robust Multiple Space Object Tracking". In: *AIAA SPACE Forum*. American Institute of Aeronautics and Astronautics, Aug. 2014.

[192] G. Tommei, A. A. Milani, and A. Rossi. "Orbit determination of space debris: admissible regions". In: *Celestial Mechanics and Dynamical Astronomy* 97.4 (2007), pp. 289–304.

[194] D. A. Vallado. *Fundamentals of Astrodynamics and Applications*. 2nd. Space Technology Library. Springer Netherlands, 2001.

[200] J. Vinti and G. Der. *Orbital and Celestial Mechanics*. Vol. 177. Progress in Astronautics and Aeronautics. American Institute of Aeronautics & Astronautics, 1998.

6

Concluding Remarks

6.1 Conclusions

This dissertation has presented a series of novel methods for solving the multi-object estimation problem using non-standard settings. When dealing with challenging scenarios, the proposed filters tackle fundamental problems of nonlinearity and non-Gaussianity of processes, high state dimensionality, high numbers of targets, statistical dependence between target states, and degenerate cases of low signal-to-noise ratio, high uncertainty, lowly observable states or uninformative observations. The proposed algorithms constitute stochastic filters, each of which is formulated to address specific aspects of the challenges at hand while offering tools to achieve compromises in conflicting goals of the studied problems.

The first algorithm, the Gauss-Hermite filter, is based on a hybrid method that combines a particle-based representation of the prior state uncertainty with an efficient grid-based method to estimate the posterior probability density. The method makes use of the prior Monte Carlo empirical measure to induce a probability mass function that approximates the posterior probability measure. This probability mass function enables accurate numerical integration, by means of the Gauss-Hermite quadrature, to compute the state estimate and covariance matrix. The filter was shown to be suitable to tackle the estimation problem for nonlinear and/or non-Gaussian processes and measurement models, including cases when the posterior densities are not unimodal. In terms of complexity, the filter requires a careful study to determine whether the added complexity will revert to performance benefit to justify its use.

The second method, the Stochastic Particle Flow, constitutes a filtering framework that addresses well-known shortcomings of sequential Monte Carlo methods when applied to nonlinear high-dimensional filtering problems. This novel method uses a Monte Carlo procedure to generate a sequence of equally-weighted samples that each guide a local solution of the Fökker-Planck equation. Using these local approximations, a mixture is produced that approximates the filtering density. The result is a statistically-sound general-purpose class of algorithms. In the context of a simple, though not trivial, high-dimensional inference problem and in comparison with state-of-the-art algorithms, the proposed approach has been shown to offer significant improvement in statistical consistency with commensurate computational expense.

The third method presented, the JPDA-EHM3, consists of a scalable multi-target tracking scheme that considers dependence between target states and enables track management by inferring existence probabilities. The proposed algorithm models target dependence by probabilistic trees on which joint probability distributions of adjacent targets are calculated. The method avoids maintaining a joint probability distribution over all the target states, which is infeasible except when the number of targets is small, and enables a scalable algorithm for

scenarios with moderately high number of targets. The numerical experiment presented made it clear that the proposed filter reduces incidence of track swapping and substantially improves tracking capability when information sensible for association disambiguation are occasionally available.

The fourth method, the Discrete-Gamma CPHD, involves a new filter that is second-order in target number, where the targets' state is assumed to follow an independent and identically distributed cluster process with the cardinality distribution modeled as a discretized Gamma distribution. The strategy employed was to mimic, based on a discrete-Gamma distribution, the procedure of a Kalman filter for the cardinality random variable, i.e., estimating sufficient statistics. As demonstrated by the numerical examples, the discrete Gamma distribution allows simple calculations for approximating the first- and second-order moments of the posterior cardinality distribution, and efficiently addresses tracking scenarios with underdispersed and slightly overdispersed target counts, without the restrictions required by the binomial filter. The results also demonstrate that the DG-CPHD is more computationally efficient than the standard CPHD implementation, especially for scenarios where a large number of CPHD cardinality terms is necessary, i.e., in situations where the number of target-generated measurements is significantly increased as many targets appear in a scene.

Finally, a novel system has been devised to perform observation-to-observation associations for objects in the geostationary belt, with low observability due to short arc lengths covered by each observation. The framework is based on a new initial orbit determination method, derived to enable a reasonable description of nearly-geosynchronous and geostationary orbits and their uncertainties, and a procedure for statistical comparison between estimated orbits in a mixed space comprising orbital elements and measurements. Any two objects observed in different epochs can then be compared to generate likelihood values that quantify their similarity. The method partially works around the computational complexity implied in standard multiple-hypothesis filtering, and focuses on producing lists of associated objects before a full-precision orbit determination algorithm can be applied.

The proposed techniques, when analyzed from an empirical perspective, appear to constitute evidence to support the primary hypothesis of this dissertation. That is, exploring extreme and challenging cases, where the fundamental shortcomings of filtering methods are exposed, highlights mathematical principles and root causes of practical problems that can be analyzed to unveil important system characteristics and to provide insights for the design of new methods.

6.2 Future work

The analyses and techniques proposed in this dissertation can be regarded as a starting point for several avenues yet to be explored.

In particular, representing filtered probability distributions to higher accuracy could be further explored in different senses. One possible direction could exploit the hybrid filtering framework established by the Gauss-Hermite filter, as proposed in Chapter 1, to combine sequential Monte Carlo methods with highly efficient grid-based methods, by resorting to effective techniques for approximating high-dimensional integrals (e.g., [113]), adopting sparse grids (e.g., Smolyak grid [182]), and scaling and positioning grids adaptively based on spectral properties of the distributions involved. Another direction could capitalize on recent ideas of partitioning

the state variables, under the Monte Carlo representation of probability measures, so that the dependence between the estimation error and dimensionality is mitigated [160].

In the same ethos of that employed by the Stochastic Particle Flow technique developed in Chapter 2, there is a vast realm in the continuum mechanics, optimal transport, and differential geometry literature that could be investigated to give rise to new methodologies for Bayesian estimation. Some instances of this line of research can be found in the works by Fred Daum [49, 47], Reich [161, 162], Moselhy [141], but the field is still in early development. Progress in this direction is of great theoretical value, especially for promoting a better understanding of geometrical and algebraic structures involved in the space of probability distributions, and a thorough characterization of operations and transformations acting on these structures that arise in filtering. To illustrate this point, we point out the extremely important but underexplored connections between Lie algebras and filtering [90], and between differential geometry and filtering [17]. There is also a great practical value in these new methodologies since, as evidenced in Chapter 2, filtering algorithms with better regularity properties are required in order to provide consistent estimates for increasingly complex phenomena.

Another avenue that can potentially follow from this dissertation is concerned with concise representations of multi-object interactions. By simply replacing the structure used to represent dependencies between targets in the JPDA-EHM3 tracker developed in Chapter 3, other filters could be derived. For instance, instead of using a discrete probability distribution of components (representing the identities of components in a mixture density), written as a dependency graph, a Gaussian graphical model could be used, where each node is a Gaussian mixture and the dependencies are modeled as conditional Gaussian densities, such that the model can be updated by Gaussian belief propagation. This description enables a more direct interpretation of the covariances between different targets, and the dependence graphical model would have a clear correspondence to the multi-target probability density.

Also, the complexity of multi-sensor multi-target tracking algorithms must be addressed in future research. This trend arises as a response to an increasing necessity of maintaining custody of enormous numbers of objects (hundredths to thousands) in applications that are very computationally demanding such as, for example, reliably keeping track of decommissioned (artificial) satellites and debris, or tracking aircraft in highly congested airspaces. The method used to derive the DG-CPHD filter in Chapter 4 opens up possibilities of new cardinality models for approximating the CPHD and decreasing the complexity of CPHD-based filters. This is very much in the same direction as that of some filters proposed recently (e.g., [173]). The reasoning for reducing complexity also applies to methods that might follow from the measurement-to-measurement technique proposed in Chapter 5, where new effective ways of associating objects can be derived by modifying some of the assumptions used in this dissertation.

Last, in the context of Chapter 5, there is a potential for a novel class of estimators and smoothers to be developed by resorting to functional and/or pseudo-spectral descriptions of trajectories (e.g., [103]), where mathematical concepts from traditional functional analysis and tools from functional data analysis can be readily borrowed.

References

[17] D. Brigo, B. Hanzon, and F. LeGland. "A differential geometric approach to nonlinear filtering: the projection filter". In: *IEEE Transactions on Automatic Control* 43.2 (Feb. 1998), pp. 247–252.

[47] F. Daum and J. Huang. "Particle flow for nonlinear filters, Bayesian decisions and transport". In: *FUSION 2013, Proceedings of the 16th International Conference on Information Fusion*. 2013, pp. 1072–1079.

[49] F. Daum and J. Huang. "Nonlinear filters with log-homotopy". In: *Proc. SPIE 6699, Signal and Data Processing of Small Targets 2007*. Vol. 6699. Signal and Data Processing of Small Targets, 2007. San Diego, California, Sept. 2007, pp. 669918–669918-15.

[90] M. Hazewinkel and S. I. Marcus. "On Lie algebras and finite dimensional filtering". In: *Stochastics* 7.1-2 (1982), pp. 29–62.

[103] B. A. Jones, A. Doostan, and G. H. Born. "Nonlinear Propagation of Orbit Uncertainty Using Non-Intrusive Polynomial Chaos". In: *Journal of Guidance, Control, and Dynamics* 36.2 (Jan. 2013), pp. 430–444.

[113] D. Krieg and E. Novak. "A Universal Algorithm for Multivariate Integration". In: *Foundations of Computational Mathematics* 17.4 (Aug. 2017), pp. 895–916.

[141] T. A. E. Moselhy and Y. M. Marzouk. "Bayesian inference with optimal maps". In: *Journal of Computational Physics* 231.23 (2012), pp. 7815–7850.

[160] P. Rebeschini and R. van Handel. "Can local particle filters beat the curse of dimensionality?" In: *Annals of Applied Probability* 25.5 (Oct. 2015), pp. 2809–2866.

[161] S. Reich. "A dynamical systems framework for intermittent data assimilation". English. In: *BIT Numerical Mathematics* 51.1 (2011), pp. 235–249.

[162] S. Reich. "A Guided Sequential Monte Carlo Method for the Assimilation of Data into Stochastic Dynamical Systems". English. In: *Recent Trends in Dynamical Systems*. Ed. by A. Johann et al. Vol. 35. Springer Proceedings in Mathematics & Statistics. Springer Basel, 2013, pp. 205–220.

[173] I. Schlangen et al. "A Second-Order PHD Filter With Mean and Variance in Target Number". In: *IEEE Transactions on Signal Processing* 66.1 (Jan. 2018), pp. 48–63.

[182] S. Smolyak. "Quadrature and interpolation formulas for tensor products of certain classes of functions". In: *Soviet Mathematics, Doklady* 4 (1963), pp. 240–243.

A

Discretization of Linear State-space Models

This appendix gives a brief account, in the perspective of the author, of an established method to discretize linear state space models. This method is well documented in the Control Engineering literature (e.g., in [150]), however, with few exceptions [123, 121, 204, 5], it is not thoroughly discussed in the multi-target tracking literature and it appears to be the case that the community would benefit from the content provided herein. The appendix provides a simple derivation of the procedure to discretize a continuous-time linear state-space model. The intent is clarifying how discrete-time models widely used in multi-target and multi-sensor tracking can be obtained from continuous-time models, by resorting to transformations in the Laplace domain and other techniques for solving stochastic differential equations.

Discretization

The problem we want to address is transforming a continuous-time linear model, given its state differential equation, into a discrete-time linear model given by a difference equation. Suppose the following continuous-time (linear time-invariant) state-space model

$$\frac{d\mathrm{x}(t)}{dt} = \mathrm{A}\mathrm{x}(t) + \mathrm{B}\mathrm{u}(t) + \mathrm{G}\mathrm{w}(t), \; t \geq 0, \tag{A.1}$$

where x is an $d_x \times 1$ state vector, A is the $d_x \times d_x$ state-transition matrix, u is an $d_u \times 1$ input vector, B is an $d_x \times d_u$ input matrix, and $\mathrm{G} \cdot \mathrm{w}$ is a perturbation term. In this appendix both B and G are assumed time-invariant for brevity of exposition. The intent is to acquire an equivalent difference equation to model a discrete-time model according to

$$\begin{aligned} \mathrm{x}(kT) &= \mathrm{A}_\mathrm{d}\mathrm{x}((k-1)T) + \mathrm{B}_\mathrm{d}\mathrm{u}(kT) + \mathrm{G}_\mathrm{d}\mathrm{w}(kT), \\ \mathrm{x}_k &= \mathrm{A}_\mathrm{d}\mathrm{x}_{k-1} + \mathrm{B}_\mathrm{d}\mathrm{u}_k + \mathrm{G}_\mathrm{d}\mathrm{w}_k, \; k \in \mathbb{N}_0, \end{aligned} \tag{A.2}$$

where T is the sampling period. Many mathematical tools are available to perform this discretization (e.g., Z-transform), but we resort to a simple technique. Let us obtain the solution for the homogeneous differential equation

$$\frac{d\mathrm{x}(t)}{dt} = \mathrm{A} \cdot \mathrm{x}(t), \; t \geq 0. \tag{A.3}$$

Applying the Laplace transform to equation (A.3) with $\mathrm{x}(0_-)$ as the state at $t = 0_-$, we get

$$\begin{aligned}\mathcal{L}\{\dot{\mathrm{x}}\,(t)\}(s) &= \mathcal{L}\{\mathrm{A}\cdot\mathrm{x}\,(t)\}(s),\\ s\cdot\mathrm{X}(s) - \mathrm{x}(0_-) &= \mathrm{A}\cdot\mathrm{X}(s),\\ \mathrm{X}(s) &= (s\cdot\mathbb{I}_{d_x} - \mathrm{A})^{-1}\mathrm{x}(0_-),\end{aligned} \tag{A.4}$$

where $\mathrm{X}(s) = \mathcal{L}\{\mathrm{x}\,(t)\}(s)$ is the Laplace transform of the state vector, $s \in \mathbb{C}$ is the frequency variable, and $\mathbb{I}_{d_x}$ is the d_x-dimensional identity matrix. By applying the inverse Laplace transform to (A.4) and using the property $\mathbb{I}_{d_x} e^{\rho} \equiv e^{\rho\mathbb{I}_{d_x}}$ for any scalar $\rho \in \mathbb{R}$, it is possible to obtain the state vector as

$$\begin{aligned}\mathrm{x}\,(t) &= \mathcal{L}^{-1}\left\{(s\cdot\mathbb{I}_{d_x} - \mathrm{A})^{-1}\mathrm{x}(0_-)\right\}(t) = \mathcal{L}^{-1}\left\{(s\cdot\mathbb{I}_{d_x} - \mathrm{A})^{-1}\mathbb{I}_{d_x}\mathrm{x}(0_-)\right\}(t)\\ &= \int_0^\infty (s\cdot\mathbb{I}_{d_x} - \mathrm{A})^{-1}\mathbb{I}_{d_x}\mathrm{x}(0_-)e^{s\cdot t}ds = \int_0^\infty (s\cdot\mathbb{I}_{d_x} - \mathrm{A})^{-1}e^{s\mathbb{I}_{d_x}\cdot t}\mathrm{x}(0_-)ds\\ &= \left[\int_0^\infty (r\mathbb{I}_{d_x})^{-1}e^{(r\mathbb{I}_{d_x}+\mathrm{A})\cdot t}dr\right]\mathrm{x}(0_-) = \left[\int_0^\infty (r\mathbb{I}_{d_x})^{-1}e^{r\mathbb{I}_{d_x}\cdot t}dr\right]e^{\mathrm{A}\cdot t}\mathrm{x}(0_-)\\ &= \left[\int_0^\infty r^{-1}e^{r\cdot t}dr\right]\mathbb{I}_{d_x}e^{\mathrm{A}\cdot t}\mathrm{x}(0_-) = \mathcal{L}^{-1}\left\{r^{-1}\right\}(t)\,\mathbb{I}_{d_x}e^{\mathrm{A}\cdot t}\mathrm{x}(0_-) = H\,(t)\,e^{\mathrm{A}\cdot t}\mathrm{x}(0_-),\\ \therefore \mathrm{x}\,(t) &= e^{\mathrm{A}\cdot t}\mathrm{x}(0_-),\ t \geq 0,\end{aligned} \tag{A.5}$$

with the variable substitution $r\mathbb{I}_{d_x} = s\mathbb{I}_{d_x} - \mathrm{A}$, $r \in \mathbb{C}$, and where the inverse Laplace transform of $1/r$ results in the Heaviside unit-step function $\mathcal{L}^{-1}\{1/r\}(t) = H(t)$. In (A.5), by choosing a specific instant $t = k\cdot T$, we have

$$\begin{aligned}\mathrm{x}(kT) &= e^{\mathrm{A}\cdot kT}\mathrm{x}(0_-) = e^{+\mathrm{A}\cdot T-\mathrm{A}\cdot T}e^{\mathrm{A}\cdot kT}\mathrm{x}(0_-) = e^{\mathrm{A}\cdot T}\overbrace{e^{\mathrm{A}\cdot(k-1)T}\mathrm{x}(0_-)}^{\mathrm{x}((k-1)T)}\\ &= e^{\mathrm{A}\cdot T}\mathrm{x}((k-1)\,T),\\ \mathrm{x}_k &= e^{\mathrm{A}\cdot T}\mathrm{x}_{k-1},\end{aligned} \tag{A.6}$$

where the correspondence $\mathrm{A_d} \equiv e^{\mathrm{A}\cdot T}$ becomes evident. Now consider the inhomogeneous differential equation

$$\frac{d\mathrm{x}\,(t)}{dt} = \mathrm{A}\mathrm{x}\,(t) + \mathrm{B}\mathrm{u}\,(t)\,,\ t \geq 0. \tag{A.7}$$

This equation could be solved by the Laplace transform method as well, but a simpler procedure provides the answer promptly. We premultiply equation (A.7) by $e^{-\mathrm{A}\cdot t}$, rearrange and integrate

it for $t \geq 0$, to get

$$\begin{aligned}
e^{-\mathrm{A}\cdot t}\dot{\mathrm{x}}\left(t\right) &= e^{-\mathrm{A}\cdot t}\mathrm{A}\mathrm{x}\left(t\right) + e^{-\mathrm{A}\cdot t}\mathrm{B}\mathrm{u}\left(t\right), \\
e^{-\mathrm{A}\cdot t}\dot{\mathrm{x}}\left(t\right) - e^{-\mathrm{A}\cdot t}\mathrm{A}\mathrm{x}\left(t\right) &= e^{-\mathrm{A}\cdot t}\mathrm{B}\mathrm{u}\left(t\right), \\
e^{-\mathrm{A}\cdot t}\dot{\mathrm{x}}\left(t\right) + \frac{d}{dt}\left(e^{-\mathrm{A}\cdot t}\right)\mathrm{x}\left(t\right) &= e^{-\mathrm{A}\cdot t}\mathrm{B}\mathrm{u}\left(t\right), \\
\frac{d}{dt}\left(e^{-\mathrm{A}\cdot t}\mathrm{x}\left(t\right)\right) &= e^{-\mathrm{A}\cdot t}\mathrm{B}\mathrm{u}\left(t\right), \\
\int_0^t \frac{d}{d\tau}\left(e^{-\mathrm{A}\cdot \tau}\mathrm{x}\left(\tau\right)\right)d\tau &= \int_0^t e^{-\mathrm{A}\cdot \tau}\mathrm{B}\mathrm{u}\left(\tau\right)d\tau, \\
e^{-\mathrm{A}\cdot t}\mathrm{x}\left(t\right) - e^{-\mathrm{A}\cdot 0}\mathrm{x}(0_-) &= \int_0^t e^{-\mathrm{A}\cdot \tau}\mathrm{B}\mathrm{u}\left(\tau\right)d\tau, \\
e^{-\mathrm{A}\cdot t}\mathrm{x}\left(t\right) &= \mathrm{x}(0_-) + \int_0^t e^{-A\cdot \tau}B\cdot u\left(\tau\right)d\tau, \\
\mathrm{x}\left(t\right) &= e^{\mathrm{A}\cdot t}\mathrm{x}(0_-) + e^{\mathrm{A}\cdot t}\int_0^t e^{-\mathrm{A}\cdot \tau}\mathrm{B}\mathrm{u}\left(\tau\right)d\tau.
\end{aligned} \tag{A.8}$$

If we evaluate (A.8) at $t = k\cdot T$, an induction similar to that in (A.6) can be used to produce

$$\begin{aligned}
&\mathrm{x}(kT) = e^{\mathrm{A}\cdot kT}\mathrm{x}(0_-) + e^{\mathrm{A}\cdot kT}\int_0^{kT} e^{-\mathrm{A}\cdot \tau}\mathrm{B}\mathrm{u}\left(\tau\right)d\tau \\
&= e^{\mathrm{A}\cdot kT}\mathrm{x}(0_-) + e^{\mathrm{A}\cdot kT}\int_0^{(k-1)T} e^{-\mathrm{A}\cdot \tau}\mathrm{B}\mathrm{u}\left(\tau\right)d\tau + e^{\mathrm{A}\cdot kT}\int_{(k-1)T}^{kT} e^{-\mathrm{A}\cdot \tau}\mathrm{B}\mathrm{u}\left(\tau\right)d\tau \\
&= e^{\mathrm{A}\cdot T}\underbrace{\left[e^{\mathrm{A}\cdot (k-1)T}\mathrm{x}(0_-) + e^{\mathrm{A}\cdot (k-1)T}\int_0^{(k-1)T} e^{-\mathrm{A}\cdot \tau}\mathrm{B}\mathrm{u}\left(\tau\right)d\tau\right]}_{\mathrm{x}((k-1)T)} + e^{\mathrm{A}\cdot kT}\int_{(k-1)T}^{kT} e^{-\mathrm{A}\cdot \tau}\mathrm{B}\mathrm{u}\left(\tau\right)d\tau,
\end{aligned}$$

$$\begin{aligned}
\therefore \mathrm{x}(kT) &= e^{\mathrm{A}\cdot T}\mathrm{x}\left(\left(k-1\right)T\right) + \overbrace{e^{\mathrm{A}\cdot kT}\int_{(k-1)T}^{kT} e^{-\mathrm{A}\cdot \tau}\mathrm{B}\mathrm{u}\left(\tau\right)d\tau}^{\mathrm{B_d u}(kT)}, \\
\mathrm{x}_k &= \mathrm{A_d x}_{k-1} + \mathrm{B_d u}_k.
\end{aligned} \tag{A.9}$$

The second term in the right-hand side of (A.9) can be recognized as the discrete-time equivalent of the input term, $\mathrm{B_d u}\left(kT\right)$. It is very common to assume that the input signal is piecewise constant (zero-order hold approximation), i.e., $\mathrm{u}\left(t\right) \coloneqq \mathrm{u}_k$ for $t \in [(k-1)\,T, kT]$, to calculate the integral in (A.9) as

$$\begin{aligned}
\mathrm{B_d u}\left(kT\right) &\equiv e^{\mathrm{A}\cdot kT}\int_{(k-1)T}^{kT} e^{-\mathrm{A}\cdot \tau}\mathrm{B}\mathrm{u}\left(\tau\right)d\tau \\
&= \int_{(k-1)T}^{kT} e^{-\mathrm{A}\cdot (\tau - kT)}\mathrm{B}\mathrm{u}_k d\tau = \int_0^T e^{\mathrm{A}\cdot \eta}\mathrm{B}\mathrm{u}_k d\eta \\
&= \mathrm{A}^{-1}\left[\int_0^T \mathrm{A}\cdot e^{\mathrm{A}\cdot \eta}d\eta\right]\mathrm{B}\mathrm{u}_k = \mathrm{A}^{-1}\left[\int_0^T \frac{d}{d\eta}\left(e^{\mathrm{A}\cdot \eta}\right)d\eta\right]\mathrm{B}\mathrm{u}_k \\
&= \mathrm{A}^{-1}\left[e^{\mathrm{A}\cdot \eta}\right]_{\eta=0}^{\eta=T}\cdot \mathrm{B}\mathrm{u}_k = \mathrm{A}^{-1}(e^{\mathrm{A}\cdot T} - \mathbb{I}_{d_x})\mathrm{B}\mathrm{u}_k,
\end{aligned} \tag{A.10}$$

where the variable substitution $\eta = -\tau + kT$ was made.

Now we shall extend the procedure for a stochastic state process $\{\mathrm{x}(t)\}_{t\geq 0}$, that is, a state model that evolves according to a stochastic differential equation similar to equation (A.7), given by

$$\begin{aligned} d\mathrm{x}(t) &= \mathrm{A}\mathrm{x}(t)dt + \mathrm{G}d\beta_t, \\ \frac{d\mathrm{x}\,(t)}{dt} &= \mathrm{A}\mathrm{x}\,(t) + \mathrm{G}\mathrm{w}_t, \end{aligned} \tag{A.11}$$

where $\{\beta_t\}_{t\geq 0}$ is a multidimensional Brownian motion process with covariance matrix specified as $\mathbb{E}[d\beta_t d\beta_t^{\mathrm{T}}] = \mathrm{Q}dt$, and $\{\mathrm{w}_t\}_{t>0}$ is a multidimensional Gaussian process such that the following correspondence holds [101]

$$\mathrm{w}_t dt \sim d\beta_t. \tag{A.12}$$

From now on $\int_T d\beta_t$ shall be interpreted as a Wiener integral over a time interval with length T, and (A.11) must be recognized as an abuse of notation since the Gaussian process is not integrable with probability one [101]. If we proceed the discretization exactly as in (A.8) and (A.9), but considering $\mathrm{G}\cdot\mathrm{w}_t$ in place of $\mathrm{B}\cdot\mathrm{u}(t)$, we get

$$\begin{aligned} \mathrm{x}(kT) &= e^{\mathrm{A}\cdot T}\mathrm{x}\left((k-1)\,T\right) + e^{\mathrm{A}\cdot kT}\int_{(k-1)T}^{kT} e^{-\mathrm{A}\cdot\tau}\mathrm{G}\mathrm{w}_\tau d\tau \\ &= e^{\mathrm{A}\cdot T}\mathrm{x}\left((k-1)\,T\right) + \int_{(k-1)T}^{kT} e^{-\mathrm{A}\cdot(\tau-kT)}\mathrm{G}\mathrm{w}_\tau d\tau. \end{aligned} \tag{A.13}$$

Equation (A.13) is simply the result of substituting $\mathrm{B}\cdot\mathrm{u}(t)$ by $\mathrm{G}\cdot\mathrm{w}_t$ in equation (A.9). Here a problem arises: the second term in the right-hand side of (A.13) is stochastic (non-deterministic) because $\{\mathrm{w}_t\}_{t\geq 0}$ is a random process, hence the integral must be defined in the mean square sense, as a stochastic integral either in the Îto or Stratonovich formalism (see [101] for details). Expressing the stochastic term requires obtaining its statistical moments. In order to compute the statistical moments of the stochastic integral in (A.13) we invoke the following properties of the Brownian motion process.

Theorem A.1 (*Theorem 4.1* in [101])**.** *Suppose a Brownian motion process $\{\beta_t\}_{t\geq 0}$ with variance $\mathbb{E}\left[d\beta_t^2\right] = \sigma^2 dt$. Let two random functions $f(t)$ and $g(t)$ satisfy:*

i) *$f(t)$ and $g(t)$ are independent of $\{\beta_{t_j} - \beta_{t_i} : (k-1)T \leq t_i \leq t \leq t_j \leq kT\}$ for all $t \in [(k-1)T, kT]$ and any $k \in \mathbb{N}_0$,*

ii) *$\int_T \mathbb{E}\left[|f(t)|^2\right] dt$, $\int_T \mathbb{E}\left[|g(t)|^2\right] dt < \infty$.*

Then

$$\mathbb{E}\left[\int_T f(t)d\beta_t\right] = \mathbb{E}\left[\int_T g(t)d\beta_t\right] = 0, \tag{A.14}$$

$$\mathbb{E}\left[\int_T f(t)d\beta_t \cdot \int_T g\,(t)\,d\beta_t\right] = \sigma^2 \int_T \mathbb{E}\left[f\,(t)\,g\,(t)\right] dt. \tag{A.15}$$

The proof of *Theorem 4.1* in [101] can be found in Chapter 4, Section 3, pages 97 to 102. As identified in (A.12), $\mathrm{w}_t dt \sim d\beta_t$, thus the mean of $\int_{(k-1)T}^{kT} e^{-\mathrm{A}\cdot(\tau-kT)}\mathrm{G}\mathrm{w}_\tau d\tau$ can be calculated

by

$$\mathbb{E}\left[\int_{(k-1)T}^{kT} e^{-\mathrm{A}\cdot(\tau-kT)}\mathrm{G}\mathrm{w}_\tau d\tau\right] = \mathbb{E}\left[\int_{(k-1)T}^{kT} e^{-\mathrm{A}\cdot(\tau-kT)}\mathrm{G}\, d\beta_\tau\right] = 0, \tag{A.16}$$

and its covariance matrix can be calculated by

$$\begin{aligned}
\mathrm{Q_d} &= \mathbb{E}\left[\left(\int_{(k-1)T}^{kT} e^{-\mathrm{A}\cdot(\tau-kT)}\mathrm{G}\mathrm{w}_\tau d\tau\right)\cdot\left(\int_{(k-1)T}^{kT} e^{-\mathrm{A}\cdot(\tau'-kT)}\mathrm{G}\mathrm{w}_{\tau'} d\tau'\right)^{\mathrm{T}}\right] \\
&= \mathbb{E}\left[\left(\int_{(k-1)T}^{kT} e^{-\mathrm{A}\cdot(\tau-kT)}\mathrm{G}\, d\beta_\tau\right)\cdot\left(\int_{(k-1)T}^{kT} e^{-\mathrm{A}\cdot(\tau'-kT)}\mathrm{G}\, d\beta_{\tau'}\right)^{\mathrm{T}}\right] \\
&= \int_{(k-1)T}^{kT} \mathbb{E}\left[\int_{(k-1)T}^{kT} e^{-\mathrm{A}\cdot(\tau-kT)}\mathrm{G}\left(d\beta_\tau d\beta_{\tau'}^{\mathrm{T}}\right)\mathrm{G}^{\mathrm{T}} e^{-\mathrm{A}^{\mathrm{T}}\cdot(\tau'-kT)}\right] \\
&= \int_{(k-1)T}^{kT} \mathbb{E}\left[e^{-\mathrm{A}\cdot(\tau-kT)}\mathrm{G}\cdot\mathrm{Q}\cdot\mathrm{G}^{\mathrm{T}} e^{-\mathrm{A}^{\mathrm{T}}\cdot(\tau-kT)}\right] d\tau \\
&= \int_{(k-1)T}^{kT} e^{-\mathrm{A}\cdot(\tau-kT)}\mathrm{G}\mathrm{Q}\mathrm{G}^{\mathrm{T}} e^{-\mathrm{A}^{\mathrm{T}}\cdot(\tau-kT)} d\tau \\
&= \int_0^T e^{\mathrm{A}\cdot\nu}\mathrm{G}\mathrm{Q}\mathrm{G}^{\mathrm{T}} e^{\mathrm{A}^{\mathrm{T}}\cdot\nu} d\nu,
\end{aligned} \tag{A.17}$$

where we made the variable substitution $\nu = -\tau + kT$.

Summary

The following equivalences of parameters between the continuous and discrete-time domains are identified:

$$\mathrm{A_d} = e^{\mathrm{A}\cdot T}, \quad \mathrm{B_d} = \mathrm{A}^{-1}(e^{\mathrm{A}\cdot T} - \mathbb{I}_{d_x})\mathrm{B}, \quad \mathrm{Q_d} = \int_0^T e^{\mathrm{A}\cdot\nu}\mathrm{G}\mathrm{Q}\mathrm{G}^{\mathrm{T}} e^{\mathrm{A}^{\mathrm{T}}\cdot\nu} d\nu. \tag{A.18}$$

Approximations

In a number of practical cases, computing the discrete-time parameters via (A.18) cannot be performed exactly. The most common approximation is of first order according to

$$\mathrm{A_d} \approx (\mathbb{I}_{d_x} + \mathrm{A}\cdot T), \quad \mathrm{B_d} \approx \mathrm{B}\cdot T, \quad \mathrm{Q_d} \approx \int_0^T (\mathbb{I}_{d_x} + A\cdot\nu)\mathrm{G}\mathrm{Q}\mathrm{G}^{\mathrm{T}}(\mathbb{I}_{d_x} + A\cdot\nu)^{\mathrm{T}} d\nu. \tag{A.19}$$

However, often, the approximated integral in (A.19) may result in a coarse approximation, which evokes the need for alternative methods to compute $\mathrm{Q_d}$. A well known technique is based on Van Loan's method [196] for evaluating integrals involving the matrix exponential. This method proposes computing the exponential of a block triangular matrix and then combining the resulting submatrices to evaluate the integral at hand. To apply Van Loan's method for

evaluating $\mathrm{Q_d}$, one must calculate (or approximate)

$$\begin{pmatrix} \mathrm{U}_2(T) & \mathrm{V}_2(T) \\ 0 & \mathrm{U}_3(T) \end{pmatrix} = \exp\left\{ \begin{pmatrix} -\mathrm{A}^{\mathrm{T}} & \mathrm{GQG}^{\mathrm{T}} \\ 0 & \mathrm{A} \end{pmatrix} T \right\}, \tag{A.20}$$

and then use $\mathrm{V}_2(T)$ and $\mathrm{U}_3(T)$ to give

$$\mathrm{Q_d} = \mathrm{U}_3^{\mathrm{T}}(T) \cdot \mathrm{V}_2(T). \tag{A.21}$$

Other alternatives involve computing $\mathrm{Q_d}$ by solving an associated Lyapunov equation for the continuous-time model [204, 5], or partioning the problem into structured subproblems and then solving them either analytically or using a system of Lyapunov and Sylvester equations [204].

Example A.2. *Nearly-constant velocity model.* Consider the continuous-time nearly-constant velocity model as

$$\begin{aligned} \dot{\mathrm{x}} &= \mathrm{Ax} + \mathrm{Gw}_t, \\ \begin{pmatrix} \dot{p} \\ \dot{v} \end{pmatrix} &= \mathrm{A} \begin{pmatrix} p \\ v \end{pmatrix} + \mathrm{G} \begin{pmatrix} 0 \\ w_{\delta v} \end{pmatrix}, \end{aligned} \tag{A.22}$$

where the state process equation in terms of the Gaussian (white) process $\{\mathrm{w}_t\}_{t>0}$ is an abuse of notation (no physical meaning) and should be understood as an Îto's SDE, $d\mathrm{x} = \mathrm{Ax}dt + \mathrm{G}d\beta_t$, for a Brownian motion $\{\beta_t\}_{t>0}$. In (A.22), p is the position, v the velocity, and

$$\mathrm{A} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \mathrm{G} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}, \mathrm{Q} = \begin{pmatrix} 0 & 0 \\ 0 & \sigma_{\delta v}^2 \end{pmatrix}, \tag{A.23}$$

where $\sigma_{\delta v}^2 = \mathbb{E}[w_{\delta v}^2]$ in view of a Gaussian (white) process with $\mathbb{E}\left[\mathrm{w}_t \mathrm{w}_s^{\mathrm{T}}\right] = \mathrm{Q}(t)\delta(t-s)$. The usual approximate discretization can be expressed by

$$\mathrm{A_d} = e^{\mathrm{A}\cdot T} \approx (\mathbb{I}_{d_x} + \mathrm{A}\cdot T) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} T = \begin{pmatrix} 1 & T \\ 0 & 1 \end{pmatrix}, \tag{A.24}$$

and

$$\begin{aligned} \mathrm{Q}_d &= \int_0^T e^{\mathrm{A}\cdot\nu} \mathrm{GQG}^{\mathrm{T}} e^{\mathrm{A}^{\mathrm{T}}\cdot\nu} d\nu \approx \int_0^T (\mathbb{I}_{d_x} + A\cdot\nu)\mathrm{GQG}^{\mathrm{T}}(\mathbb{I}_{d_x} + A\cdot\nu)^{\mathrm{T}} d\nu, \\ &= \int_0^T \begin{pmatrix} 1 & \nu \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 \\ 0 & \sigma_{\delta v}^2 \end{pmatrix} \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ \nu & 1 \end{pmatrix} d\nu \\ &= \int_0^T \begin{pmatrix} \sigma_{\delta v}^2 \nu^2 & \sigma_{\delta v}^2 \nu \\ \sigma_{\delta v}^2 \nu & \sigma_{\delta v}^2 \end{pmatrix} d\nu = \left[\begin{pmatrix} \sigma_{\delta v}^2 \nu^3/3 & \sigma_{\delta v}^2 \nu^2/2 \\ \sigma_{\delta v}^2 \nu^2/2 & \sigma_{\delta v}^2 \nu \end{pmatrix} \right]_{\nu=0}^{\nu=T} = \begin{pmatrix} T^3/3 & T^2/2 \\ T^2/2 & T \end{pmatrix} \sigma_{\delta v}^2, \end{aligned} \tag{A.25}$$

which is reminiscent of the usual discretized form of a nearly-constant velocity model.

Example A.3. *Orbital model.* Consider Lagrange's planetary equations as follows:

$$\dot{a} = \frac{2}{na}\frac{\partial R}{\partial \ell}, \tag{A.26}$$

$$\dot{e} = \frac{1-e^2}{na^2 e}\left(\frac{\partial R}{\partial \ell} - (1-e^2)^{-\frac{1}{2}}\frac{\partial R}{\partial \omega}\right), \tag{A.27}$$

$$\dot{\imath} = \frac{1}{na^2(1-e^2)^{\frac{1}{2}}}\left(\cot \imath \frac{\partial R}{\partial \omega} - \csc \imath \frac{\partial R}{\partial \Omega}\right), \tag{A.28}$$

$$\dot{\Omega} = \frac{\csc \imath}{na^2(1-e^2)^{\frac{1}{2}}}\frac{\partial R}{\partial \imath}, \tag{A.29}$$

$$\dot{\omega} = \frac{(1-e^2)^{\frac{1}{2}}}{na^2 e}\frac{\partial R}{\partial e} - \frac{\cot \imath}{na^2(1-e^2)^{\frac{1}{2}}}\frac{\partial R}{\partial \imath}, \tag{A.30}$$

$$\dot{\ell} = n - \frac{2}{na}\frac{\partial R}{\partial a} - \frac{1-e^2}{na^2 e}\frac{\partial R}{\partial e}, \tag{A.31}$$

where

a: semi-major axis,

e: (first) eccentricity,

$\imath$: inclination,

Ω: right ascension of the ascending node,

ω: argument of perigee,

ℓ: perturbed mean anomaly,

n: mean motion,

R: disturbing (potential) force function.

We write (A.26)–(A.31) in terms of the state vector $\mathrm{x} = (a, e, \imath, \Omega, \omega, \ell)^{\mathrm{T}}$ (orbital elements), where the unperturbed motion is modeled by a deterministic state (vector) function, $\mathrm{f} : \mathbb{R}^{d_x} \to \mathbb{R}^{d_x}$, $d_x = 6$, and the perturbation is modeled as a stochastic term, i.e.,

$$d\mathrm{x} = \mathrm{f}(\mathrm{x})dt + \mathrm{Q}^{\frac{1}{2}} d\mathrm{w}_t,\ t \geq 0, \tag{A.32}$$

where $\{\mathrm{w}_t\}_{t\geq 0}$, $\mathrm{w}_t \in \mathbb{R}^{d_x}$, is a standard multivariate Wiener process. The state function is given by

$$\mathrm{f}(\mathrm{x}) = \begin{pmatrix} f_1(\mathrm{x}) \\ f_2(\mathrm{x}) \\ f_3(\mathrm{x}) \\ f_4(\mathrm{x}) \\ f_5(\mathrm{x}) \\ f_6(\mathrm{x}) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ n \end{pmatrix} \equiv \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \mu^{\frac{1}{2}} a^{-\frac{3}{2}} \end{pmatrix}, \tag{A.33}$$

where μ is Earth's gravitational constant, and the mean motion is identified as $n = \mu^{\frac{1}{2}} a^{-\frac{3}{2}}$. The continuous-time covariance matrix Q depends on the perturbation vector that accounts for

the motion uncertainty as

$$\mathrm{g(x)}dt = \begin{pmatrix} \sigma_{\delta a}(t) \\ \sigma_{\delta e}(t) \\ \sigma_{\delta i}(t) \\ \sigma_{\delta \Omega}(t) \\ \sigma_{\delta \omega}(t) \\ \sigma_{\delta \ell}(t) \end{pmatrix} dt \triangleq \begin{pmatrix} \frac{2}{na}\frac{\partial R}{\partial \ell} \\ \frac{1-e^2}{na^2 e}\left(\frac{\partial R}{\partial \ell} - (1-e^2)^{-\frac{1}{2}}\frac{\partial R}{\partial \omega}\right) \\ \frac{1}{na^2(1-e^2)^{\frac{1}{2}}}\left(\cot i \frac{\partial R}{\partial \omega} - \csc i \frac{\partial R}{\partial \Omega}\right) \\ \frac{(1-e^2)^{\frac{1}{2}}}{na^2 e}\frac{\partial R}{\partial e} - \frac{\cot i}{na^2(1-e^2)^{\frac{1}{2}}}\frac{\partial R}{\partial i} \\ \frac{(1-e^2)^{\frac{1}{2}}}{na^2 e}\frac{\partial R}{\partial e} - \frac{\cot i}{na^2(1-e^2)^{\frac{1}{2}}}\frac{\partial R}{\partial i} \\ -\frac{2}{na}\frac{\partial R}{\partial a} - \frac{1-e^2}{na^2 e}\frac{\partial R}{\partial e} \end{pmatrix} dt \sim \mathrm{Q}^{\frac{1}{2}} d\mathrm{w}_t. \tag{A.34}$$

Linearizing the continuous-time state function in the neighborhood of $\mathrm{x}_k = \mathrm{x}(t_k)$ we have

$$\mathrm{f(x)} \approx \mathrm{Ax} + \mathrm{B} = \begin{pmatrix} 1 & & & \dots & & 0 \\ 0 & 1 & & & & \\ & & 1 & & & \\ \vdots & & & 1 & & \vdots \\ & & & & 1 & \\ -\frac{3n_k}{2a_k} & & & \dots & 0 & 1 \end{pmatrix} \begin{pmatrix} a \\ e \\ i \\ \Omega \\ \omega \\ \ell \end{pmatrix} + \begin{pmatrix} 0 \\ \\ \\ \vdots \\ \\ \frac{5}{2}n_k \end{pmatrix}, \; t \geq t_k, \tag{A.35}$$

as a consequence of

$$\begin{aligned} f_6(\mathrm{x}) = f_6(a) &\approx f_6(a_k) + \frac{\partial}{\partial a} f_6(a_k)(a - a_k) = \mu^{\frac{1}{2}} a_k^{-\frac{3}{2}} + \frac{\partial}{\partial a}(\mu^{\frac{1}{2}} a^{-\frac{3}{2}})|_{a=a_k}(a - a_k) \\ &= \mu^{\frac{1}{2}} a_k^{-\frac{3}{2}} - \frac{3}{2}\mu^{\frac{1}{2}} a_k^{-\frac{3}{2}-1}(a - a_k) = \mu^{\frac{1}{2}} a_k^{-\frac{3}{2}} + \frac{3}{2}\mu^{\frac{1}{2}} a_k^{-\frac{3}{2}} - \frac{3}{2}\mu^{\frac{1}{2}} a_k^{-\frac{3}{2}-1} \cdot a \\ &= \frac{5}{2}\mu^{\frac{1}{2}} a_k^{-\frac{3}{2}} - \frac{3}{2}\mu^{\frac{1}{2}} a_k^{-\frac{3}{2}-1} \cdot a \\ &= \frac{5}{2} n_k - \frac{3}{2}\frac{n_k}{a_k} \cdot a. \end{aligned}$$

Applying (A.19), we obtain

$$\mathrm{A_d} \approx (\mathbb{I}_{n_x} + \mathrm{A}\cdot T) = \begin{pmatrix} 1 & & & \dots & & 0 \\ 0 & 1 & & & & \\ \vdots & & 1 & & & \vdots \\ & & & 1 & & \\ & & & & 1 & \\ -\frac{3n_k}{2a_k}T & & & \dots & 0 & 1 \end{pmatrix}, \quad \mathrm{B_d} \approx \mathrm{B}\cdot T = \begin{pmatrix} 0 \\ \\ \vdots \\ \\ \\ \frac{5}{2}n_k T \end{pmatrix}, \tag{A.36}$$

$$\begin{aligned} \mathrm{Q_d} &\approx \int_0^T (\mathbb{I}_{n_x} + A\cdot\nu)\mathrm{GQG^T}(\mathbb{I}_{n_x} + A\cdot\nu)^{\mathrm{T}} d\nu \\ &= \begin{pmatrix} \sigma_{\delta a}^2 T & 0 & & \dots & 0 & \sigma_{\delta a}^2\left(-\frac{3n_k}{2a_k}\right)\frac{T^2}{2} \\ 0 & \sigma_{\delta e}^2 T & & & & 0 \\ \vdots & & \sigma_{\delta i}^2 T & & & \\ & & & \sigma_{\delta\Omega}^2 T & & \vdots \\ 0 & & & & \sigma_{\delta\omega}^2 T & 0 \\ \sigma_{\delta a}^2\left(-\frac{3n_k}{2a_k}\right)\frac{T^2}{2} & 0 & \dots & & 0 & \sigma_{\delta\ell}^2 T + \sigma_{\delta a}^2\left(\frac{3n_k}{2a_k}\right)^2\frac{T^3}{3} \end{pmatrix}. \end{aligned} \tag{A.37}$$

This appendix presented the derivation of a traditional procedure to discretize continuous-time, stochastic linear state space models. The discrete-time model derived in *Examples A.2* and *A.3* are of practical utility in this dissertation. The discrete-time model obtained in *Example A.2* is used in numerical examples of Chapters 2–4 and the discrete-time model obtained in *Example A.3* is used in the numerical example of the Chapter 5.

References

[5] P. Axelsson and F. Gustafsson. "Discrete-Time Solutions to the Continuous-Time Differential Lyapunov Equation With Applications to Kalman Filtering". In: *IEEE Transactions on Automatic Control* 60.3 (Mar. 2015), pp. 632–643.

[101] A. H. Jazwinski. *Stochastic Processes and Filtering Theory*. New York: Academic Press, 1970.

[121] X. R. Li and V. P. Jilkov. "Survey of Maneuvering Target Tracking. Part II: Motion Models of Ballistic and Space Targets". In: *IEEE Transactions on Aerospace and Electronic Systems* 46.1 (Jan. 2010), pp. 96–119.

[123] X. R. Li and V. P. Jilkov. "Survey of maneuvering target tracking. Part I. Dynamic models". In: *IEEE Transactions on Aerospace and Electronic Systems* 39.4 (Oct. 2003), pp. 1333–1364.

[150] K. Ogata. *Modern Control Engineering*. 4th. Upper Saddle River, NJ, USA: Prentice Hall PTR, 2001.

[196] C. Van Loan. "Computing integrals involving the matrix exponential". In: *IEEE Transactions on Automatic Control* 23.3 (June 1978), pp. 395–404.

[204] N. Wahlström, P. Axelsson, and F. Gustafsson. "Discretizing stochastic dynamical systems using Lyapunov equations". In: *IFAC Proceedings Volumes* 47.3 (2014). 19th IFAC World Congress, pp. 3726–3731.

Bibliography

[1] D. L. Alspach and H. W. Sorenson. "Nonlinear Bayesian estimation using Gaussian sum approximations". In: *IEEE Transactions on Automatic Control* 17.4 (Aug. 1972), pp. 439–448.

[2] B. D. O. Anderson and J. B. Moore. *Optimal Filtering*. Englewood Cliffs, NJ: Prentice-Hall, 1979.

[3] T. M. Apostol. "An Elementary View of Euler's Summation Formula". In: *The American Mathematical Monthly* 106.5 (1999), pp. 409–418.

[4] M. S. Arulampalam et al. "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking". In: *IEEE Transactions on Signal Processing* 50.2 (2002), pp. 174–188.

[5] P. Axelsson and F. Gustafsson. "Discrete-Time Solutions to the Continuous-Time Differential Lyapunov Equation With Applications to Kalman Filtering". In: *IEEE Transactions on Automatic Control* 60.3 (Mar. 2015), pp. 632–643.

[6] Y. Bar-Shalom, F. Daum, and J. Huang. "The Probabilistic Data Association Filter". In: *IEEE Control Systems* 29.6 (Dec. 2009), pp. 82–100.

[7] Y. Bar-Shalom, P. K. Willett, and X. Tian. *Tracking and Data Fusion: A Handbook of Algorithms*. YBS Publishing, Apr. 2011.

[8] R. H. Battin. *An Introduction to the Mathematics and Methods of Astrodynamics*. AIAA education series. American Institute of Aeronautics & Astronautics, 1999.

[9] J.-D. Benamou and Y. Brenier. "A computational fluid mechanics solution to the Monge-Kantorovich mass transfer problem". In: *Numerische Mathematik* 84.3 (2000), pp. 375–393.

[10] V. E. Beneš. "Exact finite-dimensional filters for certain diffusions with nonlinear drift". In: *Stochastics* 5.1-2 (1981), pp. 65–92.

[11] N. Bergman. "Recursive Bayesian Estimation: Navigation and Tracking Applications". PhD thesis. Linköping University, Department of Electrical Engineering, 1999.

[12] A. Beskos et al. "Error Bounds and Normalising Constants for Sequential Monte Carlo Samplers in High Dimensions". In: *Advances in Applied Probability* 46.1 (2014), pp. 279–306.

[13] M. Betancourt. "A General Metric for Riemannian Manifold Hamiltonian Monte Carlo". In: *Geometric Science of Information: First International Conference, GSI 2013, Paris, France, August 28-30, 2013. Proceedings*. Ed. by F. Nielsen and F. Barbaresco. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 327–334.

[14] P. Bickel, B. Li, and T. Bengtsson. "Sharp failure rates for the bootstrap particle filter in high dimensions". In: *Pushing the Limits of Contemporary Statistics: Contributions in Honor of Jayanta K. Ghosh*. Ed. by B. Clarke and S. Ghosal. Vol. 3. Collections. Beachwood, Ohio, USA: Institute of Mathematical Statistics, 2008, pp. 318–329.

[15] H. A. P. Blom and E. A. Bloem. "Joint Particle Filtering of Multiple Maneuvering Targets From Unassociated Measurements". In: *Journal of Advances in Information Fusion* 1.1 (2006), pp. 15–34.

[16] H. A. P. Blom and E. A. Bloem. "Probabilistic data association avoiding track coalescence". In: *IEEE Transactions on Automatic Control* 45.2 (Feb. 2000), pp. 247–259.

[17] D. Brigo, B. Hanzon, and F. LeGland. "A differential geometric approach to nonlinear filtering: the projection filter". In: *IEEE Transactions on Automatic Control* 43.2 (Feb. 1998), pp. 247–252.

[18] D. S. Bryant et al. "The CPHD Filter With Target Spawning". In: *IEEE Transactions on Signal Processing* 65.5 (Mar. 2017), pp. 13124–13138.

[19] R. Bucy. "Nonlinear filtering theory". In: *IEEE Transactions on Automatic Control* 10.2 (Apr. 1965), pp. 198–198.

[20] R. S. Bucy and P. D. Joseph. *Filtering for Stochastic Processes with Applications to Guidance*. Ed. by A. C. Publishing. 2nd. John Wiley & Sons, 1968.

[21] R. S. Bucy and K. D. Senne. "Digital Synthesis of Non-linear Filters". In: *Automatica* 7.3 (May 1971), pp. 287–298.

[22] A. Budhiraja and G. Kallianpur. "Approximations to the solution of the Zakai equation using multiple Wiener and Stratonovich integral expansions". In: *Stochastics and Stochastic Reports* 56.3-4 (1996), pp. 271–315.

[23] A. Budhiraja and G. Kallianpur. "The Feynman-Stratonovich semigroup and Stratonovich integral expansions in nonlinear filtering". English. In: *Applied Mathematics and Optimization* 35.1 (1997), pp. 91–116.

[24] P. Bunch and S. Godsill. "Particle filtering with progressive Gaussian approximations to the optimal importance density". In: *IEEE 5th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), 2013*. 2013, pp. 360–363.

[25] P. Bunch and S. Godsill. "Approximations of the Optimal Importance Density using Gaussian Particle Flow Importance Sampling". In: *Journal of the American Statistical Association* 111.514 (2016), pp. 748–762.

[26] O. Cappé, S. J. Godsill, and É. Moulines. "An Overview of Existing Methods and Recent Advances in Sequential Monte Carlo". In: *Proceedings of the IEEE* 95.5 (May 2007), pp. 899–924.

[27] S. Chakraborty and D. Chakravarty. "Discrete Gamma Distributions: Properties and Parameter Estimations". In: *Communications in Statistics - Theory and Methods* 41.18 (2012), pp. 3301–3324.

[28] S. Challa, Y. Bar-Shalom, and V. Krishnamurthy. "Nonlinear filtering via generalized Edgeworth series and Gauss-Hermite quadrature". In: *IEEE Transactions on Signal Processing* 48.6 (2000), pp. 1816–1820.

[29] S. Challa et al. "Fundamentals of Object Tracking". In: Cambridge books online. Cambridge University Press, 2011. Chap. 5, pp. 161–167.

[30] S. Choi et al. "Discussion and application of the homotopy filter". In: *Proc. SPIE. 8050, Signal Processing, Sensor Fusion, and Target Recognition XX*. Vol. 8050. May 2011, pp. 805021–805021-12.

[31] C. Chow and C. Liu. "Approximating Discrete Probability Distributions with Dependence Trees". In: *IEEE Transactions on Information Theory* 14.3 (May 1968), pp. 462–467.

[32] P. C. Consul and G. C. Jain. "A Generalization of the Poisson Distribution". In: *Technometrics* 15.4 (1973), pp. 791–799.

[33] P. Consul, S. Kotz, and F. Famoye. *Lagrangian Probability Distributions*. Birkhäuser Boston, 2006.

[34] J. Cornebise, É. Moulines, and J. Olsson. "Adaptive methods for sequential importance sampling with application to state space models". In: *Statistics and Computing* 18.4 (Dec. 2008), pp. 461–480.

[35] D. Crisan. "Particle Filters in a Continuous Time Framework". In: *IEEE Nonlinear Statistical Signal Processing Workshop, 2006*. Sept. 2006, pp. 73–78.

[36] D. Crisan, P. Del Moral, and T. J. Lyons. "Interacting Particle Systems Approximations of the Kushner-Stratonovitch Equation". English. In: *Advances in Applied Probability* 31.3 (1999), pp. 819–838.

[37] D. Crisan and K. Li. "Generalised particle filters with Gaussian mixtures". In: *Stochastic Processes and their Applications* 125.7 (2015), pp. 2643–2673.

[38] D. Crisan and T. Lyons. "A particle approximation of the solution of the Kushner-Stratonovitch equation". English. In: *Probability Theory and Related Fields* 115.4 (1999), pp. 549–578.

[39] D. Crisan and J. Xiong. "Approximate McKean-Vlasov representations for a class of SPDEs". In: *Stochastics* 82 (2010), pp. 53–68.

[40] B. Dacorogna and J. Moser. "On a partial differential equation involving the Jacobian determinant". In: *Annales de l'I.H.P. Analyse non linéaire* 7.1 (1990), pp. 1–26.

[41] A. S. Dalalyan. "Theoretical guarantees for approximate sampling from smooth and log-concave densities". In: *ArXiv e-prints* (Dec. 2014). arXiv: 1412.7392 [stat.CO].

[42] A. S. Dalalyan and A. B. Tsybakov. "Sparse Regression Learning by Aggregation and Langevin Monte-Carlo". In: *Journal of Computing and System Sciences* 78.5 (Sept. 2012), pp. 1423–1443.

[43] A. S. Dalalyan. "Theoretical guarantees for approximate sampling from smooth and log-concave densities". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* (2016).

[44] D. J. Daley and D. Vere-Jones. *An Introduction to the Theory of Point Processes*. Vol. Volume I: Elementary Theory and Methods. Probability and its Applications. New York, Berlin, Paris: Springer, 2003.

[45] D. J. Daley and D. Vere-Jones. *An Introduction to the Theory of Point Processes*. Vol. Volume II: General Theory and Structure. Probability and its Applications. New York, Berlin, Paris: Springer, 2008.

[46] F. Daum and J. Huang. "Curse of dimensionality and particle filters". In: *IEEE Aerospace Conference, 2003*. Vol. 4. Mar. 2003, pp. 1979–1993.

[47] F. Daum and J. Huang. "Particle flow for nonlinear filters, Bayesian decisions and transport". In: *FUSION 2013, Proceedings of the 16th International Conference on Information Fusion*. 2013, pp. 1072–1079.

[48] F. Daum and J. Huang. "Generalized particle flow for nonlinear filters". In: *Proc. SPIE 7698, Signal and Data Processing of Small Targets 2010*. Vol. 7698. Orlando, Florida, Apr. 2010, pp. 76980I–76980I-12.

[49] F. Daum and J. Huang. "Nonlinear filters with log-homotopy". In: *Proc. SPIE 6699, Signal and Data Processing of Small Targets 2007*. Vol. 6699. Signal and Data Processing of Small Targets, 2007. San Diego, California, Sept. 2007, pp. 669918–669918-15.

[50] F. Daum and J. Huang. "Nonlinear filters with particle flow". In: *Proc. SPIE 7445, Signal and Data Processing of Small Targets 2009*. Vol. 7445. Aug. 2009, 74450R–74450R-9.

[51] F. Daum and J. Huang. "Particle flow for nonlinear filters with log-homotopy". In: *Proc. SPIE 6969, Signal and Data Processing of Small Targets 2008*. Vol. 6969. Apr. 2008, pp. 696918–696918-12.

[52] F. Daum and J. Huang. "Particle flow with non-zero diffusion for nonlinear filters, Bayesian decisions and transport". In: *Proc. SPIE 8857, Signal and Data Processing of Small Targets 2013*. Vol. 8745. Sept. 2013, 87450P–87450P-13.

[53] F. Daum, J. Huang, and A. Noushin. "Exact particle flow for nonlinear filters". In: *Proc. SPIE 7697, Signal Processing, Sensor Fusion, and Target Recognition XIX*. Vol. 7697. Apr. 2010, pp. 769704–769704-19.

[54] F. E. Daum. "Exact finite-dimensional nonlinear filters". In: *IEEE Transactions on Automatic Control* 31.7 (July 1986), pp. 616–622.

[55] P. Del Moral. *Feynman-Kac Formulae: Genealogical and Interacting Particle Systems with Applications*. New York: Springer, 2004.

[56] P. Del Moral, A. Doucet, and A. Jasra. "Sequential Monte Carlo samplers". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 68.3 (2006), pp. 411–436.

[57] P. Del Moral and J. Houssineau. "Particle Association Measures and Multiple Target Tracking". In: *Theoretical Aspects of Spatial-Temporal Modeling*. Ed. by G. W. Peters and T. Matsui. Tokyo: Springer Japan, 2015, pp. 1–30.

[58] E. Delande, J. Houssineau, and D. E. Clark. "Multi-object filtering with stochastic populations". In: *ArXiv e-prints* (Jan. 2015). arXiv: 1501.04671 [stat.AP].

[59] E. Delande et al. “Regional Variance for Multi-Object Filtering”. In: *IEEE Transactions on Signal Processing* 62.13 (July 2014), pp. 3415–3428.

[60] K. J. DeMars, M. K. Jah, and P. W. Schumacher. “Initial Orbit Determination using Short-Arc Angle and Angle Rate Data”. In: *IEEE Transactions on Aerospace and Electronic Systems* 48.3 (July 2012), pp. 2628–2637.

[61] K. J. DeMars and M. K. Jah. “Probabilistic Initial Orbit Determination Using Gaussian Mixture Models”. In: *Journal of Guidance, Control, and Dynamics* 36.5 (July 2013), pp. 1324–1335.

[62] J. Deutscher, A. Blake, and I. Reid. “Articulated body motion capture by annealed particle filtering”. In: *IEEE Conference on Computer Vision and Pattern Recognition, 2000*. Vol. 2. 2000, pp. 126–133.

[63] T. Ding and M. J. Coates. “Implementation of the Daum-Huang exact-flow particle filter”. In: *IEEE Statistical Signal Processing Workshop (SSP)*. 2012, pp. 257–260.

[64] A. Doucet, S. Godsill, and C. Andrieu. “On sequential Monte Carlo sampling methods for Bayesian filtering”. In: *Statistics and Computing* 10.3 (July 2000), pp. 197–208.

[65] A. Doucet and A. M. Johansen. “A Tutorial on Particle Filtering and Smoothing: Fifteen Years Later”. In: *The Oxford Handbook of Nonlinear Filtering*. Ed. by D. Crisan and B. Rozovsky. Oxford University Press, 2009.

[66] A. Doucet and A. M. Johansen. “A Tutorial on Particle Filtering and Smoothing: Fifteen Years Later”. In: Oxford Handbook of Nonlinear Filtering. 2011, pp. 656–704.

[67] S. Duane et al. “Hybrid Monte Carlo”. In: *Physics Letters B* 195.2 (1987), pp. 216–222.

[68] O. Erdinc, P. Willett, and Y. Bar-Shalom. “Probability Hypothesis Density Filter for Multitarget Multisensor Tracking”. In: *FUSION 2005, Proceedings of the 8th International Conference on Information Fusion*. Vol. 1. July 2005, p. 8.

[69] P. R. Escobal. *Methods of Orbit Determination*. R. E. Krieger Publishing Company, 1976.

[70] L. C. Evans and W. Gangbo. “Differential equations methods for the Monge-Kantorovich mass transfer problem”. In: *Memoirs of the American Mathematical Society* 137 (1999), p. 653.

[71] D. Farnocchia et al. “Innovative methods of correlation and orbit determination for space debris”. In: *Celestial Mechanics and Dynamical Astronomy* 107.1 (2010), pp. 169–185.

[72] K. Ferentinos. “On Tchebycheff’s type inequalities”. In: *Trabajos de Estadística y de Investigación Operativa* 33.1 (1982), pp. 125–132.

[73] T. E. Fortmann, Y. Bar-Shalom, and M. Scheffe. “Multi-target tracking using joint probabilistic data association”. In: *19th IEEE Conference on Decision and Control including the Symposium on Adaptive Processes*. Dec. 1980, pp. 807–812.

[74] K. Fujimoto and D. J. Scheeres. “Non-linear Bayesian orbit determination based on the generalized admissible region”. In: *FUSION 2012, Proceedings of the 15th International Conference on Information Fusion*. July 2012, pp. 2043–2049.

[75] K. Fujimoto et al. “Association of Short-Arc Optical Tracks via the Direct Bayesian Admissible Region: Theory and Application”. In: *6th European Conference on Space Debris*. Vol. 723. ESA Special Publication. Aug. 2013, p. 67.

[76] T. Fukushima. “Fast Procedure Solving Universal Kepler’s Equation”. In: *Celestial Mechanics and Dynamical Astronomy* 75.3 (Nov. 1999), pp. 201–226.

[77] S. M. Gadaleta, J. T. Horwood, and A. B. Poore. “Short arc gating in multiple hypothesis tracking for space surveillance”. In: *Proc. SPIE 8385, Sensors and Systems for Space Applications V, 83850Y*. Vol. 8385. May 2012, 83850Y–83850Y-15.

[78] C. W. Gardiner. *Stochastic methods: A Handbook for the Natural and Social Sciences*. 2nd. Springer-Verlag, Berlin–Heidelberg–New York–Tokyo, 1985.

[79] A. Gelman and X.-L. Meng. “Simulating normalizing constants: from importance sampling to bridge sampling to path sampling”. In: *Statistical Science* 13.2 (May 1998), pp. 163–185.

[80] M. Girolami and B. Calderhead. “Riemann manifold Langevin and Hamiltonian Monte Carlo methods”. In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 73.2 (2011), pp. 123–214.

[81] J. Goldberger, S. Gordon, and H. Greenspan. “An efficient image similarity measure based on approximations of KL-divergence between two Gaussian mixtures”. In: *Proceedings of the Ninth IEEE International Conference on Computer Vision*. Vol. 1. Oct. 2003, pp. 487–493.

[82] R. H. Gooding. “A new procedure for the solution of the classical problem of minimal orbit determination from three lines of sight”. In: *Celestial Mechanics and Dynamical Astronomy* 66.4 (1996), pp. 387–423.

[83] N. J. Gordon, D. J. Salmond, and A. F. M. Smith. “Novel approach to nonlinear/non-Gaussian Bayesian state estimation”. In: *IEEE Proceedings - Radar and Signal Processing* 140.2 (Apr. 1993), pp. 107–113.

[84] R. Graham. “Statistical Theory of Instabilities in Stationary Nonequilibrium Systems with Applications to Lasers and Nonlinear Optics”. In: *Springer Tracts in Modern Physics: Ergebnisse der exakten Naturwissenschaftenc; Volume 66*. Ed. by G. Höhler. Berlin, Heidelberg: Springer Berlin Heidelberg, 1973, pp. 1–97.

[85] G. F. Gronchi, L. Dimare, and A. Milani. “Orbit determination with the two-body integrals”. In: *Celestial Mechanics and Dynamical Astronomy* 107.3 (July 2010), pp. 299–318.

[86] C. E. Gutiérrez. *The Monge-Ampère Equation*. Progress in Nonlinear Differential Equations and Their Applications. Birkhäuser Boston, 2001.

[87] J. Hagmar et al. “Optimal parameterization of posterior densities using homotopy”. In: *FUSION 2011, Proceedings of the 14th International Conference on Information Fusion*. July 2011, pp. 1–8.

[88] U. D. Hanebeck and J. Steinbring. “Progressive Gaussian filtering based on Dirac Mixture approximations”. In: *FUSION 2012, Proceedings of the 15th International Conference on Information Fusion*. July 2012, pp. 1697–1704.

[89] U. D. Hanebeck, K. Briechle, and A. Rauh. "Progressive Bayes: a new framework for nonlinear state estimation". In: *Proc. SPIE. 5099, Multisensor, Multisource Information Fusion: Architectures, Algorithms, and Applications 2003*. Vol. 5099. Apr. 2003, pp. 256–267.

[90] M. Hazewinkel and S. I. Marcus. "On Lie algebras and finite dimensional filtering". In: *Stochastics* 7.1-2 (1982), pp. 29–62.

[91] J. Heng, A. Doucet, and Y. Pokern. "Gibbs Flow for Approximate Transport with Applications to Bayesian Computation". In: *ArXiv e-prints* (Sept. 2015). arXiv: 1509.08787 [stat.CO].

[92] N. Higham. "Computing the nearest correlation matrix–a problem from finance". In: *IMA Journal of Numerical Analysis* 22 (2002), pp. 329–343.

[93] F. R. Hoots and R. L. Roehrich. *Spacetrack Report #3: Models for Propagation of the NORAD Element Sets*. Tech. rep. Colorado Springs, CO: U.S. Air Force Aerospace Defense Command, 1980.

[94] P. R. Horridge and S. Maskell. "A scalable method of tracking targets with dependent distributions". In: *FUSION 2009, Proceedings of the 12th International Conference on Information Fusion*. July 2009, pp. 603–610.

[95] P. R. Horridge and S. Maskell. "Real-Time Tracking Of Hundreds Of Targets With Efficient Exact JPDAF Implementation". In: *FUSION 2006, Proceedings of the 9th International Conference on Information Fusion*. July 2006, pp. 1–8.

[96] I. Hussein et al. "Probabilistic Admissible Region for Short-Arc Angles-only Observations". In: *AMOS Conference*. 2014.

[97] K. Ito and K. Xiong. "Gaussian filters for nonlinear filtering problems". In: *IEEE Transactions on Automatic Control* 45.5 (May 2000), pp. 910–927.

[98] K. Ito and K. Xiong. "Gaussian Filters for Nonlinear Filtering Problems". In: *IEEE Transactions on Automatic Control* 45 (2000), pp. 910–927.

[99] P. Jäeckel. *A Note on Multivariate Gauss-Hermite Quadrature*. May 2005. URL: http://jaeckel.16mb.com/ANoteOnMultivariateGaussHermiteQuadrature.pdf.

[100] H.-K. Janssen. "On a Lagrangean for classical field dynamics and renormalization group calculations of dynamical critical properties". English. In: *Zeitschrift für Physik B Condensed Matter* 23.4 (1976), pp. 377–380.

[101] A. H. Jazwinski. *Stochastic Processes and Filtering Theory*. New York: Academic Press, 1970.

[102] B. Jia, M. Xin, and Y. Cheng. "Sparse-grid Quadrature Nonlinear Filtering". In: *Automatica* 48.2 (Feb. 2012), pp. 327–341.

[103] B. A. Jones, A. Doostan, and G. H. Born. "Nonlinear Propagation of Orbit Uncertainty Using Non-Intrusive Polynomial Chaos". In: *Journal of Guidance, Control, and Dynamics* 36.2 (Jan. 2013), pp. 430–444.

[104] B. A. Jones, B.-T. Vo, and B.-N. Vo. "Generalized Labeled Multi-Bernoulli Space-Object Tracking with Joint Prediction and Update". In: *SPACE Conferences and Exposition*. American Institute of Aeronautics and Astronautics, Sept. 2016.

[105] R. Jordan and D. Kinderlehrer. "An Extended Variational Principle". In: *Partial Differential Equations and Applications: Collected Papers in Honor of Carlo Pucci*. Ed. by G. Talenti, E. Vesentini, and P. Marcellini. Lecture Notes in Pure and Applied Mathematics. Taylor & Francis, 1996. Chap. 18.

[106] R. Jordan, D. Kinderlehrer, and F. Otto. "The Variational Formulation of the Fokker–Planck Equation". In: *SIAM Journal on Mathematical Analysis* 29.1 (1998), pp. 1–17.

[107] S. J. Julier and J. K. Uhlmann. "Unscented Filtering and Nonlinear Estimation". In: *Proceedings of the IEEE*. 2004, pp. 401–422.

[108] R. E. Kalman. "A New Approach to Linear Filtering and Prediction Problems". In: *Transactions of the ASME–Journal of Basic Engineering* 82.D (1960), pp. 35–45.

[109] M. Khan and M. Ulmke. "Improvements in the implementation of log-homotopy based particle flow filters". In: *FUSION 2015, Proceedings of the 18th International Conference on Information Fusion*. July 2015, pp. 74–81.

[110] M. Klaas, N. de Freitas, and A. Doucet. "Toward Practical N^2 Monte Carlo: the Marginal Particle Filter". In: *ArXiv e-prints* (July 2012). arXiv: 1207.1396 [stat.CO].

[111] J. H. Kotecha and P. M. Djurić. "Gaussian particle filtering". In: *IEEE Transactions on Signal Processing* 51 (2003), pp. 2592–2601.

[112] J. H. Kotecha and P. M. Djurić. "Gaussian sum particle filtering". In: *IEEE Transactions on Signal Processing* 51.10 (2003), pp. 2602–2612.

[113] D. Krieg and E. Novak. "A Universal Algorithm for Multivariate Integration". In: *Foundations of Computational Mathematics* 17.4 (Aug. 2017), pp. 895–916.

[114] H. J. Kushner. "On the differential equations satisfied by conditional probability densities of Markov processes, with applications". In: *Journal of the SIAM, Series A (Control)* 2.1 (1964), pp. 106–119.

[115] H. J. Kushner and A. S. Budhiraja. "A Nonlinear Filtering Algorithm Based on an Approximation of the Conditional Distribution". In: *IEEE Transactions on Automatic Control* 45 (1999), pp. 580–585.

[116] H. Lamb. *Hydrodynamics*. University Press, 1895.

[117] R. A. Lau and J. L. Williams. "A structured mean field approach for existence-based multiple target tracking". In: *FUSION 2016, Proceedings of the 19th International Conference on Information Fusion*. July 2016, pp. 1111–1118.

[118] F. Le Gland and N. Oudjane. "A Sequential Particle Algorithm that Keeps the Particle System Alive". In: Stochastic Hybrid Systems: Theory and Safety Critical Applications. Springer Berlin Heidelberg, 2006, pp. 351–389.

[119] D. A. Levin, Y. Peres, and E. L. Wilmer. "Markov chains and mixing times". In: With a chapter on coupling from the past by James G. Propp and David B. Wilson. Providence, R.I. American Mathematical Society, 2009. Chap. 4.

[120] H. Li. "Geostationary Satellites Collocation". In: SpringerLink: Bücher. Springer Berlin Heidelberg, 2014. Chap. 4: Geostationary Orbit Perturbation.

[121] X. R. Li and V. P. Jilkov. "Survey of Maneuvering Target Tracking. Part II: Motion Models of Ballistic and Space Targets". In: *IEEE Transactions on Aerospace and Electronic Systems* 46.1 (Jan. 2010), pp. 96–119.

[122] X. R. Li and Z. Zhao. "Measuring Estimator's Credibility: Noncredibility Index". In: *FUSION 2006, Proceedings of the 9th International Conference on Information Fusion*. July 2006, pp. 1–8.

[123] X. R. Li and V. P. Jilkov. "Survey of maneuvering target tracking. Part I. Dynamic models". In: *IEEE Transactions on Aerospace and Electronic Systems* 39.4 (Oct. 2003), pp. 1333–1364.

[124] Q. Liu and D. A. Pierce. "A note on Gauss-Hermite quadrature". In: *Biometrika* 81.3 (1994), pp. 624–629.

[125] R. P. S. Mahler. *Advances in Statistical Multisource-Multitarget Information Fusion*. Needham, MA, USA: Artech House, Inc., 2014.

[126] R. P. S. Mahler. "Multitarget Bayes Filtering via First-Order Multitarget Moments". In: *IEEE Transactions on Aerospace and Electronic Systems* 39.4 (Oct. 2003), pp. 1152–1178.

[127] R. P. S. Mahler. "A theory of PHD filters of higher order in target number". In: *Proc. SPIE 6235, Signal Processing, Sensor Fusion, and Target Recognition XV*. Vol. 6235. May 2006, 62350K–62350K-12.

[128] R. P. S. Mahler. *An Introduction to Multisource-Multitarget Statistics and Its Applications*. Technical Monograph. Eagan MN: Lockheed Martin, 2000.

[129] R. P. S. Mahler. "PHD Filters of Higher Order in Target Number". In: *IEEE Transactions on Aerospace and Electronic Systems* 43.4 (Oct. 2007), pp. 1523–1543.

[130] R. P. S. Mahler. "PHD Filters of Second Order in Target Number". In: *Proc. SPIE 6236, Signal and Data Processing of Small Targets*. Vol. 6236. May 2006, 62360P–62360P-12.

[131] R. P. S. Mahler. *Statistical Multisource-Multitarget Information Fusion*. Norwood, MA, USA: Artech House, Inc., 2007.

[132] F. L. Markley. "Kepler Equation solver". In: *Celestial Mechanics and Dynamical Astronomy* 63.1 (Mar. 1995), pp. 101–111.

[133] A. Marrs, S. Maskell, and Y. Bar-Shalom. "Expected likelihood for tracking in clutter with particle filters". In: *Proc. SPIE. 4728, Signal and Data Processing of Small Targets 2002*. Vol. 4728. Aug. 2002, pp. 230–239.

[134] J. M. Maruskin, D. J. Scheeres, and K. T. Alfriend. "Correlation of Optical Observations of Objects in Earth Orbit". In: *Journal of Guidance, Control, and Dynamics* 32.1 (Jan. 2009), pp. 194–209.

[135] S. Maskell, M. Briers, and R. Wright. "Fast Mutual Exclusion". In: *Proc. SPIE 5428, Signal and Data Processing of Small Targets 2004*. Vol. 5428. Aug. 2004, pp. 526–536.

[136] S. Maskell et al. "Tracking using a radar and a problem specific proposal distribution in a particle filter". In: *IEEE Proceedings - Radar, Sonar and Navigation*. Vol. 152. Institute of Electrical and Electronics Engineers Inc., Jan. 2005. Chap. 5, pp. 315–322.

[137] S. Meyn and R. L. Tweedie. *Markov Chains and Stochastic Stability*. 2nd. New York, NY, USA: Cambridge University Press, 2009.

[138] A. Milani. "The Asteroid Identification Problem: I. Recovery of Lost Asteroids". In: *Icarus* 137.2 (1999), pp. 269–292.

[139] A. Milani and Z. Knežević. "From Astrometry to Celestial Mechanics: Orbit Determination with Very Short Arcs". In: *Celestial Mechanics and Dynamical Astronomy* 92.1 (2005), pp. 1–18.

[140] G. N. Milstein, J. G. M. Schoenmakers, and V. Spokoiny. "Transition Density Estimation for Stochastic Differential Equations via Forward-Reverse Representations". English. In: *Bernoulli* 10.2 (2004), pp. 281–312.

[141] T. A. E. Moselhy and Y. M. Marzouk. "Bayesian inference with optimal maps". In: *Journal of Computational Physics* 231.23 (2012), pp. 7815–7850.

[142] J. Moser. "On the Volume Elements on a Manifold". In: *Transactions of the American Mathematical Society* 120.2 (1965), pp. 286–294.

[143] J. E. Moyal. "The General Theory of Stochastic Population Processes". In: *Acta Mathematica* 108 (1962), pp. 1–31.

[144] D. Musicki and R. Evans. "Joint Integrated Probabilistic Data Association - JIPDA". In: *FUSION 2002, Proceedings of the 5th International Conference on Information Fusion*. Vol. 2. July 2002, pp. 1120–1125.

[145] C. Musso, N. Oudjane, and F. Le Gland. "Improving Regularised Particle Filters". In: *Sequential Monte Carlo Methods in Practice*. Ed. by A. Doucet, N. de Freitas, and N. Gordon. Sequential Monte Carlo Methods in Practice. New York: Springer-Verlag, 2001. Chap. 12, pp. 247–271.

[146] R. M. Neal. "Annealed Importance Sampling". In: *ArXiv e-prints* (Mar. 1998).

[147] R. M. Neal. "Annealed importance sampling". English. In: *Statistics and Computing* 11.2 (2001), pp. 125–139.

[148] R. M. Neal. *Bayesian Learning for Neural Networks*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 1996.

[149] E. Novak. "Some Results on the Complexity of Numerical Integration". In: *Monte Carlo and Quasi-Monte Carlo Methods: MCQMC, Leuven, Belgium, April 2014*. Ed. by R. Cools and D. Nuyens. Springer International Publishing, 2016, pp. 161–183.

[150] K. Ogata. *Modern Control Engineering*. 4th. Upper Saddle River, NJ, USA: Prentice Hall PTR, 2001.

[151] N. Oudjane and C. Musso. "Progressive correction for regularized particle filters". In: *FUSION 2000, Proceedings of the 3rd International Conference on Information Fusion*. Vol. 2. July 2000, THB2/10–THB2/17.

[152] T. Ozaki. "A bridge between nonlinear time series models and nonlinear stochastic dynamical systems: a local linearization approach". In: *Statistica Sinica* 2.1 (1992), pp. 113–135.

[153] L. Y. Pao. "Multisensor multitarget mixture reduction algorithms for tracking". In: *Journal of Guidance, Control, and Dynamics* 17.6 (Nov. 1994), pp. 1205–1211.

[154] J. Pearl. "Reverend Bayes on inference engines: a distributed hierarchical approach". In: *AAAI-82, Proceedings of the 2nd National Conference on AI*. Pittsburg, PA, 1982, pp. 133–136.

[155] N. S. Pillai, A. M. Stuart, and A. H. Thiéry. "Optimal Scaling and Diffusion Limits for the Langevin Algorithm in High Dimensions". In: *Annals of Applied Probability* 22.6 (Dec. 2012), pp. 2320–2356.

[156] M. K. Pitt and N. Shephard. "Filtering via Simulation: Auxiliary Particle Filters". In: *Journal of the American Statistical Association* 94.446 (1999), pp. 590–599.

[157] W. H. Press et al. *Numerical Recipes in C: The Art of Scientific Computing*. New York, NY, USA: Cambridge University Press, 1988.

[158] J. Ramsay and B. W. Silverman. *Functional Data Analysis*. 2nd. Springer Series in Statistics. Springer, 2005.

[159] R. C. Rao. "Information and the accuracy attainable in the estimation of statistical parameters". In: *Bulletin of Calcutta Mathematical Society* 37 (1945), pp. 81–91.

[160] P. Rebeschini and R. van Handel. "Can local particle filters beat the curse of dimensionality?" In: *Annals of Applied Probability* 25.5 (Oct. 2015), pp. 2809–2866.

[161] S. Reich. "A dynamical systems framework for intermittent data assimilation". English. In: *BIT Numerical Mathematics* 51.1 (2011), pp. 235–249.

[162] S. Reich. "A Guided Sequential Monte Carlo Method for the Assimilation of Data into Stochastic Dynamical Systems". English. In: *Recent Trends in Dynamical Systems*. Ed. by A. Johann et al. Vol. 35. Springer Proceedings in Mathematics & Statistics. Springer Basel, 2013, pp. 205–220.

[163] D. B. Reid. "An Algorithm for Tracking Multiple Targets". In: *IEEE Transactions on Automatic Control* 24.6 (Dec. 1979), pp. 843–854.

[164] S. Reuter et al. "The Labeled Multi-Bernoulli Filter". In: *IEEE Transactions on Signal Processing* 62.12 (June 2014), pp. 3246–3260.

[165] H. Risken. *The Fokker-Planck Equation: Methods of Solution and Applications*. 2nd, Springer-Verlag, 1989.

[166] B. Ristic, S. Arulampalam, and N. Gordon. *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Artech House, 2004.

[167] G. O. Roberts, A. Gelman, and W. R. Gilks. "Weak convergence and optimal scaling of random walk Metropolis algorithms". In: *Annals of Applied Probability* 7.1 (Feb. 1997), pp. 110–120.

[168] G. O. Roberts and O. Stramer. "Langevin Diffusions and Metropolis-Hastings Algorithms". English. In: *Methodology And Computing In Applied Probability* 4.4 (2002), pp. 337–357.

[169] G. O. Roberts and J. S. Rosenthal. "Optimal Scaling of Discrete Approximations to Langevin Diffusions". In: *Journal of the Royal Statistical Society. Series B (Statistical Methodology)* 60.1 (1998), pp. 255–268.

[170] A. E. Roy. *Orbital Motion*. Adam Hilger Limited, 1978.

[171] H. P. Saal, N. M. O. Heess, and S. Vijayakumar. "Multimodal Nonlinear Filtering Using Gauss-Hermite Quadrature". In: *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2011, Athens, Greece, September 5-9, 2011, Proceedings, Part III*. Ed. by D. Gunopulos et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 81–96.

[172] S. Sarkar et al. "A Kushner-Stratonovich Monte Carlo filter applied to nonlinear dynamical system identification". In: *Physica D: Nonlinear Phenomena* 270 (2014), pp. 46–59.

[173] I. Schlangen et al. "A Second-Order PHD Filter With Mean and Variance in Target Number". In: *IEEE Transactions on Signal Processing* 66.1 (Jan. 2018), pp. 48–63.

[174] D. Schuhmacher, B.-T. Vo, and B.-N. Vo. "A Consistent Metric for Performance Evaluation of Multi-Object Filters". In: *IEEE Transactions on Signal Processing* 56.8 (Aug. 2008), pp. 3447–3457.

[175] F. Septier and G. W. Peters. "Langevin and Hamiltonian Based Sequential MCMC for Efficient Bayesian Filtering in High-Dimensional Spaces". In: *IEEE Journal of Selected Topics in Signal Processing* 10.2 (Mar. 2016), pp. 312–327.

[176] G. Shmueli et al. "A useful distribution for fitting discrete data: revival of the Conway-Maxwell-Poisson distribution". In: *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 54.1 (2005), pp. 127–142.

[177] B. W. Silverman. *Density Estimation for Statistics and Data Analysis*. London: Chapman and Hall, Apr. 1986.

[178] M. Šimandl, J. Královec, and T. Söderström. "Advanced point-mass method for nonlinear state estimation". In: *Automatica* 42.7 (July 2006), pp. 1133–1145.

[179] J. A. Siminski et al. "Short-arc tracklet association for geostationary objects". In: *Advances in Space Research* 53 (Apr. 2014), pp. 1184–1194.

[180] N. Singh et al. "Multiple Hypothesis Tracking (MHT) for Space Surveillance: Results and Simulation Studies". In: *Proceedings of the 2013 Advanced Maui Optical and Space Surveillance Technologies Conference (Wailea, Maui, Hawaii)*. Sept. 2013.

[181] W. M. Smart. "Celestial Mechanics". In: London: Longmans, Green, and Co., 1953, p. 38.

[182] S. Smolyak. "Quadrature and interpolation formulas for tensor products of certain classes of functions". In: *Soviet Mathematics, Doklady* 4 (1963), pp. 240–243.

[183] C. Snyder et al. "Obstacles to High-Dimensional Particle Filtering". In: *Monthly Weather Review* 136.12 (Dec. 2008), pp. 4629–4640.

[184] E. M. Soop. *Handbook of Geostationary Orbits*. Space Technology Library. Springer Netherlands, 1994.

[185] H. W. Sorenson and D. L. Alspach. "Recursive Bayesian Estimation Using Gaussian Sums". In: *Automatica* 7.4 (July 1971), pp. 465–479.

[186] J. Stauch et al. "Mutual Application of Joint Probabilistic Data Association, Filtering, and Smoothing Techniques for Robust Multiple Space Object Tracking". In: *AIAA SPACE Forum*. American Institute of Aeronautics and Astronautics, Aug. 2014.

[187] R. L. Stratonovich. "Optimum nonlinear systems which bring about a separation of a signal with constant parameters from noise". In: *Radiofizika* 2.6 (1959), pp. 892–901.

[188] R. Streit, C. Degen, and W. Koch. "The Pointillist Family of Multitarget Tracking Filters". In: *ArXiv e-prints* (May 2015). arXiv: 1505.08000 [stat.AP].

[189] A. H. Stroud and D. Secrest. *Gaussian quadrature formulas*. Englewood Cliffs, NJ: Prentice-Hall, 1966.

[190] L. Svensson et al. "Set JPDA Filter for Multitarget Tracking". In: *IEEE Transactions on Signal Processing* 59.10 (Oct. 2011), pp. 4677–4691.

[191] G. Terejanu et al. "A novel Gaussian Sum Filter Method for accurate solution to the nonlinear filtering problem". In: *FUSION 2008, Proceedings of the 11th International Conference on Information Fusion*. June 2008, pp. 1–8.

[192] G. Tommei, A. A. Milani, and A. Rossi. "Orbit determination of space debris: admissible regions". In: *Celestial Mechanics and Dynamical Astronomy* 97.4 (2007), pp. 289–304.

[193] M. Treiber and A. Kesting. *Traffic flow dynamics: Data, Models and Simulation*. Springer-Verlag Berlin Heidelberg, 2013.

[194] D. A. Vallado. *Fundamentals of Astrodynamics and Applications*. 2nd. Space Technology Library. Springer Netherlands, 2001.

[195] R. van der Merwe et al. *The Unscented Particle Filter*. Tech. rep. CUED/F-INFENG/TR 380. Cambridge University, Engineering Department, 2000.

[196] C. Van Loan. "Computing integrals involving the matrix exponential". In: *IEEE Transactions on Automatic Control* 23.3 (June 1978), pp. 395–404.

[197] M. Venugopal, R. M. Vasu, and D. Roy. "An Ensemble Kushner-Stratonovich-Poisson Filter for Recursive Estimation in Nonlinear Dynamical Systems". In: *IEEE Transactions on Automatic Control* 61.3 (Mar. 2016), pp. 823–828.

[198] C. Vergé et al. "On parallel implementation of sequential Monte Carlo methods: the island particle model". English. In: *Statistics and Computing* 25.2 (2015), pp. 243–260.

[199] C. Villani. *Topics in Optimal Transportation*. Graduate studies in Mathematics. American Mathematical Society, 2003.

[200] J. Vinti and G. Der. *Orbital and Celestial Mechanics*. Vol. 177. Progress in Astronautics and Aeronautics. American Institute of Aeronautics & Astronautics, 1998.

[201] B.-N. Vo and W. K. Ma. "The Gaussian Mixture Probability Hypothesis Density Filter". In: *IEEE Transactions on Signal Processing* 54.11 (Nov. 2006), pp. 4091–4104.

[202] B.-T. Vo, B.-N. Vo, and A. Cantoni. "The Cardinality Balanced Multi-Target Multi-Bernoulli Filter and Its Implementations". In: *IEEE Transactions on Signal Processing* 57.2 (Feb. 2009), pp. 409–423.

[203] B.-T. Vo, B.-N. Vo, and A. Cantoni. "Analytic Implementations of the Cardinalized Probability Hypothesis Density Filter". In: *IEEE Transactions on Signal Processing* 55.7 (July 2007), pp. 3553–3567.

[204] N. Wahlström, P. Axelsson, and F. Gustafsson. "Discretizing stochastic dynamical systems using Lyapunov equations". In: *IFAC Proceedings Volumes* 47.3 (2014). 19th IFAC World Congress, pp. 3726–3731.

[205] H. W. Watson and F. Galton. "On the Probability of the Extinction of Families". In: *The Journal of the Anthropological Institute of Great Britain and Ireland* 4 (1875), pp. 138–144.

[206] J. L. Williams. "Marginal multi-Bernoulli filters: RFS derivation of MHT, JIPDA, and association-based MeMBer". In: *IEEE Transactions on Aerospace and Electronic Systems* 51.3 (July 2015), pp. 1664–1687.

[207] J. L. Williams and R. A. Lau. "Data association by loopy belief propagation". In: *FUSION 2010, Proceedings of the 13th International Conference on Information Fusion*. July 2010, pp. 1–8.

[208] J. Williams and R. Lau. "Approximate evaluation of marginal association probabilities with belief propagation". In: *IEEE Transactions on Aerospace and Electronic Systems* 50.4 (Oct. 2014), pp. 2942–2959.

[209] D. Wood. *The Computation of Polylogarithms*. Tech. rep. 15-92*. University of Kent, Canterbury, UK: University of Kent, Computing Laboratory, June 1992, pp. 182–196.

[210] C. Wu and C. Han. "Quadrature Kalman Particle Filter". In: *Journal of Systems Engineering and Electronics* 21.2 (2010), pp. 175–179.

[211] M. Zakai. "On the optimal filtering of diffusion processes". English. In: *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete* 11.3 (1969), pp. 230–243.