

Západočeská univerzita v Plzni

Fakulta Aplikovaných Věd

Katedra kybernetiky

# BAKALÁŘSKÁ PRÁCE

PLZEŇ, 2018

JAN KREJČÍ

University of West Bohemia  
Faculty of Applied Sciences  
Department of Cybernetics

## **BACHELOR THESIS**

Uncertainty propagation for tracking of moving  
objects

PILSEN, 2018

JAN KREJČÍ

ZÁPADOČESKÁ UNIVERZITA V PLZNI

Fakulta aplikovaných věd

Akademický rok: 2017/2018

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Jan KREJČÍ**

Osobní číslo: **A15B0537P**

Studijní program: **B3918 Aplikované vědy a informatika**

Studijní obor: **Kybernetika a řídicí technika**

Název tématu: **Propagace nejistoty v úloze sledování polohy pohybujících se objektů**

Zadávací katedra: **Katedra kybernetiky**

### Z á s a d y p r o v y p r a c o v á n í :

1. Seznamte se s literaturou zabývající se propagací nejistoty v úloze sledování.
2. Vyberte vhodné metody a implementujte je.
3. Porovnejte vybrané metody teoreticky a na simulačních příkladech.

Rozsah grafických prací: **dle potřeby**  
Rozsah kvalifikační práce: **30-40 stránek A4**  
Forma zpracování bakalářské práce: **tištěná**

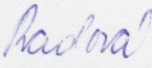
Seznam odborné literatury:

**Dodá vedoucí bakalářské práce.**

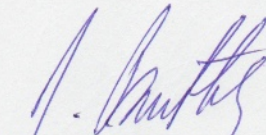
Vedoucí bakalářské práce: **Doc. Ing. Ondřej Straka, Ph.D.**  
Katedra kybernetiky

Datum zadání bakalářské práce: **1. listopadu 2017**

Termín odevzdání bakalářské práce: **18. května 2018**

  
Doc. Dr. Ing. Vlasta Radová  
děkanka



  
Prof. Ing. Josef Psutka, CSc.  
vedoucí katedry

V Plzni dne 1. listopadu 2017

## PROHLÁŠENÍ

Předkládám tímto k posouzení a obhajobě bakalářskou práci zpracovanou na závěr studia na Fakultě aplikovaných věd Západočeské univerzity v Plzni.

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně a výhradně s použitím odborné literatury a pramenů, jejichž úplný seznam je její součástí.

V plzni dne .....

.....

*vlastnoruční podpis*

## PODĚKOVÁNÍ

Tímto bych rád poděkoval panu doc. Ing. Ondřeji Strakovi, Ph.D za odborné vedení, cenné rady a připomínky, poskytnuté materiály, trpělivost, čas a ochotu při zpracovávání této bakalářské práce.

## Anotace

Tato bakalářská práce se zabývá analýzou vybraných metod využívaných v úloze propagace počáteční nejistoty pro dlouhodobé časové horizonty. Vybrané metody jsou teoreticky analyzovány a následně testovány na dvou simulačních modelech pohybu objektů po oběžné dráze Země. Výsledky simulací jsou poté porovnávány mezi sebou.

**klíčová slova:** neurčitost, propagace nejistoty, predikce, matematický model, poloha, stav, stochastický systém, oběžná dráha, linearizace

## Annotation

This bachelor thesis deals with analysis of selected methods utilized within the problem of propagating initial uncertainty for a long time horizon. First, these methods are theoretically analyzed; second, they are tested using two cases of Earth orbit movement models. The results are compared subsequently.

**keywords:** uncertainty, propagation of uncertainty, prediction, mathematical model, position, state, stochastic system, orbital mechanics, linearization

---

# Contents

<b>List of Figures</b>	<b>iv</b>
<b>Nomenclature</b>	<b>v</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 Theory Preliminaries</b>	<b>3</b>
2.1 Uncertainty Representation . . . . .	3
2.1.1 Statistical Moments . . . . .	3
2.1.2 Gaussian and Gaussian Mixture Model Distributions . . . . .	4
2.2 Splitting a Gaussian Distribution . . . . .	5
2.3 Differential Entropy of a Gaussian Distribution . . . . .	7
2.3.1 Entropy of a Linearized System State . . . . .	8
<b>3 Uncertainty Propagators</b>	<b>9</b>
3.1 Problem Statement . . . . .	11
3.1.1 Fokker-Planck Equation . . . . .	12
3.2 Monte Carlo Simulation . . . . .	13
3.2.1 The Method Description . . . . .	13
3.2.2 Analysis and Discussion . . . . .	13
3.3 LinCov . . . . .	15
3.3.1 The Method Derivation and Description . . . . .	15
3.3.2 Analysis and Discussion . . . . .	16
3.4 FOTE Based Propagation . . . . .	17
3.4.1 The Method Description . . . . .	17
3.4.2 The Alternative Derivation . . . . .	17

3.4.3	Analysis and Discussion . . . . .	18
3.5	CADET . . . . .	20
3.5.1	The Method Derivation and Description . . . . .	20
3.5.2	Analysis and Discussion . . . . .	22
3.6	UT . . . . .	23
3.6.1	The Method Description . . . . .	23
3.6.2	Analysis and Discussion . . . . .	24
3.7	AEGIS . . . . .	25
3.7.1	The Original Method Description . . . . .	25
3.7.2	Improvements of the Method . . . . .	26
3.7.3	Analysis and Discussion . . . . .	27
<b>4</b>	<b>Testing Problems</b>	<b>29</b>
4.0.1	Likelihood Agreement Between Distributions . . . . .	29
4.1	High-Earth-Orbit Test Case . . . . .	31
4.1.1	Results . . . . .	32
4.2	Low-Earth-Orbit Test Case . . . . .	37
4.2.1	Results . . . . .	39
4.3	Analysis . . . . .	44
<b>5</b>	<b>Conclusion</b>	<b>46</b>
	<b>Bibliography</b>	<b>48</b>



---

# List of Figures

2.1	Illustration of splitting a Gaussian distribution. . . . .	6
3.1	Uncertainty propagation methods outline. . . . .	10
3.2	Illustration of uncertainty propagation. . . . .	11
3.3	Monte Carlo simulation graphical illustration. . . . .	14
3.4	LinCov technique graphical illustration. . . . .	16
3.5	FOTE based propagation graphical illustration. . . . .	19
3.6	CADET method graphical illustration. . . . .	22
3.7	UT propagation graphical illustration. . . . .	24
3.8	AEGIS method propagation scheme. . . . .	28
3.9	AEGIS method, graphical illustration. . . . .	28
4.1	The testing problems graphical illustration. . . . .	30
4.2	LEO: Position and Velocity after approx. 1 orbit: Monte Carlo, LinCov, FOTE, CADET and UT propagation, 95.4% confidence ellipses. . . . .	33
4.3	HEO: Position and Velocity after approx. 1 orbit: s-AEGIS-3 . . . . .	34
4.4	HEO: Position and Velocity after approx. 1 orbit: e-AEGIS-3 . . . . .	34
4.5	LEO: Position and Velocity after approx. 2 orbits: Monte Carlo, LinCov, FOTE, CADET and UT propagation, 95.4% confidence ellipses. . . . .	35
4.6	LEO: LAM of LinCov, FOTE, CADET, UT, s-AEGIS-3 and e-AEGIS-3 with respect to MC. . . . .	35
4.7	HEO: Entropy of linear propagator and nonlinear UT propagator. . . . .	36
4.8	HEO: s-AEGIS-3 and e-AEGIS-3 number of GMM components. . . . .	36
4.9	LEO: Position and Velocity after 1 orbit: Monte Carlo, LinCov, FOTE, CADET and UT propagation, 95.4% confidence ellipses. . . . .	40

4.10 LEO: Position and Velocity after 2 orbits: Monte Carlo, LinCov, FOTE, CADET and UT propagation, 95.4% confidence ellipses. . . . .	40
4.11 LEO: Position and Velocity after 2 orbits: s-AEGIS-3 . . . . .	41
4.12 LEO: Position and Velocity after 2 orbits: e-AEGIS-3 . . . . .	41
4.13 LEO: Position and Velocity after 2 orbits: e-AEGIS-5 . . . . .	42
4.14 LEO: LAM of LinCov, FOTE, CADET, UT, s-AEGIS-3 and e-AEGIS-3,5 with respect to MC. . . . .	42
4.15 LEO: Entropy of linear propagator and nonlinear UT propagator. . . . .	43
4.16 LEO: s-AEGIS-3 and e-AEGIS-3,5 number of GMM components. . . . .	43

---

# Nomenclature

$\mathbf{A}(\mathbf{x}(t), t)$	=	derivative of the system dynamics $\mathbf{f}$ with respect to $\mathbf{x}$
$ \mathbf{A} $	=	determinant of the square matrix $\mathbf{A}$
$E[.]$	=	Expectation operator
$\mathbf{f}(\mathbf{x}(t), t)$	=	system dynamics vector evaluated at state $\mathbf{x}(t)$
$H(\mathbf{x})$	=	differential entropy of the random variable $\mathbf{x}$
$\mathcal{L}(p, q)$	=	likelihood agreement measure between distributions $p(\mathbf{x})$ , and $q(\mathbf{x})$
$\mathbf{m}$	=	mean vector of a Gaussian distribution
$\mu$	=	gravitational constant of the Earth
$\mathbf{P}$	=	covariance matrix of a Gaussian distribution
$p_g(\mathbf{x}; \mathbf{m}, \mathbf{P})$	=	Gaussian probability density function evaluated at $\mathbf{x}$
$\mathbf{r}, \dot{\mathbf{r}}$	=	Cartesian Earth-Centered Inertial (ECI) position coordinates and their time derivatives (velocity)
$\mathbf{v}, \dot{\mathbf{v}}$	=	ECI velocity coordinates and their time derivatives (acceleration)
$\mathbf{x}(t), \dot{\mathbf{x}}(t)$	=	state of the system at time $t$ and its time derivative
$\alpha_i$	=	weight of the $i$ th component in a Gaussian mixture model
$\phi$	=	state solution flow
$\Phi$	=	state transition matrix
$\lambda_i$	=	$i$ th eigenvector of a square matrix $\mathbf{A}$

# Introduction

By nature, real systems usually have some non-deterministic features which may be modeled in terms of stochastic models. If a state-space model is taken into consideration, the core element, which represents the system status is its state. The state is usually not measurable directly and therefore has to be estimated from noisy data. Consider a continuous stochastic dynamical system model with a discrete-time measurement. State of such model at any time is per se described by some probability measure. In order to study the time evolution of this state, one needs to determine its description at a desired time, which can be done within two phases: *time update* (prediction) and *measurement update* (filtering).

The *measurement update* goal is to improve the current knowledge of the system state using a single measurement at a time.

The *time update* is a continuous process which might be called *uncertainty propagation*. This process takes place between the measurement time instants and it can be very difficult if the system under consideration is highly nonlinear. In many cases of interest the state description reveals a Gaussian distribution, but it can easily become non-Gaussian, depending on chosen coordinate system and application [1, 2]. This means that the uncertainty propagation is often only an approximation of a very complex reality. Various uncertainty propagation methods have been developed in order to satisfy common objectives including computational requirements (speed), accuracy and system suitability.

Such methods and above mentioned models are widely used in areas such as automatic control, economics or the space situational awareness (SSA). In terms of SSA, a compact summary of uncertainty propagation methods can be found in [3].

The stochastic system behavior can be affected in terms of multiple error sources. Referring the seminal paper [4], the uncertainty sources can be differentiated with respect to

an epistemological perspective into two fundamental types, *aleatory* and *epistemic* uncertainty.

*Aleatory uncertainty* describes the irreducible natural uncertainty, e.g. additive noise.

*Epistemic uncertainty* describes the designer's lack of knowledge, inaccurate model e.g. uncertainty in initial condition, or uncertainty in model parameters.

The continuous dynamic systems that are theoretically studied in this work are deterministic state-space models with an uncertain initial condition. No measurements, inputs, or noise within the system state evolution are considered. The objective of this thesis is to analyze some of the uncertainty propagation tools (uncertainty propagators). The theoretical methods are implemented and tested with a chosen nonlinear model. The results are compared in various objectives settled before.

The thesis is organized as follows. First, basic theory is presented. Theoretical analysis of chosen uncertainty propagation methods follows. Two test problems are then settled with the following results. Finally, comparison and conclusion are given.

---

# Theory Preliminaries

## 2.1 Uncertainty Representation

Let a random process  $\mathbf{x}(t) \in \mathbb{R}^n$  (continuous in both time and values) be distributed by the cumulative density function (CDF)  $P(\mathbf{x}(t))$ . If the CDF is absolutely continuous, the process  $\mathbf{x}(t)$  can be described by the probability density function (PDF)  $p(\mathbf{x}(t))$ , which must be a non-negative function and also must integrate to unity over the state space. Then, the probability of  $\mathbf{x}(t)$  being in some volume  $\Xi$  can be computed as

$$\Pr(\mathbf{x}(t) \in \Xi) = \int_{\Xi} dP(\mathbf{x}(t)) = \int_{\Xi} p(\mathbf{x}(t)) d\mathbf{x} \quad (2.1)$$

For any fixed time  $t$ ,  $\mathbf{x}(t)$  is a random vector denoted simply as  $\mathbf{x} \in \mathbb{R}^n$ .

### 2.1.1 Statistical Moments

Statistical moments of a random vector  $\mathbf{x}$  describe some overall qualities of the distribution  $p(\mathbf{x})$  at a given time  $t$ . The first two moments are mean and covariance. The mean  $\mathbf{m} \in \mathbb{R}^n$  measures the central tendency, the covariance matrix  $\mathbf{P} \in \mathbb{R}^{n \times n}$  measures spread of the distribution of its mean,

$$\mathbf{m} = E[\mathbf{x}] = \int_{\mathbb{R}^n} \xi p(\xi) d\xi \quad (2.2)$$

$$\mathbf{P} = E[(\mathbf{x} - \mathbf{m})(\mathbf{x} - \mathbf{m})^T] = \int_{\mathbb{R}^n} (\xi - \mathbf{m})(\xi - \mathbf{m})^T p(\xi) d\xi \quad (2.3)$$

where  $E[\cdot]$  denotes the expectation operator. Other moments can be defined similarly. The formal derivative of mean and covariance matrix with respect to time is

$$\dot{\mathbf{m}} = E[\dot{\mathbf{x}}] \quad (2.4)$$

$$\dot{\mathbf{P}} = E[\dot{\mathbf{x}}\mathbf{x}^T + \mathbf{x}\dot{\mathbf{x}}^T] - (\dot{\mathbf{m}}\mathbf{m}^T + \mathbf{m}\dot{\mathbf{m}}^T) \quad (2.5)$$

## 2.1.2 Gaussian and Gaussian Mixture Model Distributions

If a random vector  $\mathbf{x}$  is *Gaussian*, its PDF is given by

$$p_g(\mathbf{x}; \mathbf{m}, \mathbf{P}) = \frac{1}{\sqrt{|2\pi\mathbf{P}|}} \exp\left\{-\frac{1}{2}(\mathbf{x} - \mathbf{m})^T \mathbf{P}^{-1}(\mathbf{x} - \mathbf{m})\right\} \quad (2.6)$$

where  $|\cdot|$  represents the matrix determinant,  $\mathbf{m}$  and  $\mathbf{P}$  are mean and covariance matrix, respectively. The Gaussian distribution can be described with only mean and a covariance matrix which limits the PDF to have only certain proportions: it is centered around the mean and its contours can only be ellipsoids (due to the covariance). If the true distribution  $p_{\mathbf{x}}$  cannot be sufficiently represented using the Gaussian PDF (the random vector  $\mathbf{x}$  is not Gaussian), other representations such as the *Gaussian mixture model* (GMM) can be used as follows,

$$p_{\mathbf{x}}(\mathbf{x}, \mathbf{m}, \mathbf{P}) \approx \sum_{i=1}^N \alpha_i p_g(\mathbf{x}; \mathbf{m}_i, \mathbf{P}_i) \quad (2.7)$$

where

$$\alpha_i \geq 0 \quad \forall i \in \{1, 2, \dots, N\} \quad \text{and} \quad \sum_{i=1}^N \alpha_i = 1 \quad (2.8)$$

with  $N$  being the number of components and  $\mathbf{m}_i, \mathbf{P}_i$  being mean and covariance associated with the  $i$ th component, respectively. It can be shown [5] that approximation of a PDF  $p_{\mathbf{x}}$  by a GMM PDF converges uniformly as the number of components  $N$  increases without bounds. This means that the more components of GMM are used, the better approximation is yielded, as can be seen in the AEGIS uncertainty propagation method [6], which analysis is in Section 3.7.

## 2.2 Splitting a Gaussian Distribution

If the true PDF  $p_{\mathbf{x}}$  is approximated by a GMM, the parameters of the GMM naturally change during the time evolution. In order to catch better the shape of the true PDF in a future time, the number  $N$  of the GMM components can be enlarged replacing any of the existing GMM component with several new, smaller ones (a replacing GMM).

First, consider replacing an univariate standard Gaussian distribution with a GMM consisting of  $\Delta l$  components as the following splitting process,

$$p_g(x; 0, 1) \approx \sum_{i=1}^{\Delta l} \tilde{\alpha}_i p_g(x; \tilde{m}_i, \tilde{\sigma}_i^2) \quad (2.9)$$

where  $\tilde{\alpha}_i, \tilde{m}_i, \tilde{\sigma}_i$  are the replacing GMM parameters to be found. If the replacing components are constrained to have the same variance parameter  $\tilde{\sigma}_i$ , the parameters can be found for example by minimizing the Kullback-Leibler divergence between the two distributions (the PDF to be splitted and the GMM to be used instead) which yields splitting libraries presented in Tab. 2.1 and 2.2, which are taken from [6], where the splitting process is fully described. A four-component splitting library can be found in [7].

Second, consider a splitting process in a multivariate case, such as

$$\alpha p_g(\mathbf{x}, \mathbf{m}, \mathbf{P}) \approx \sum_{i=1}^{\Delta l} \alpha_i p_g(\mathbf{x}; \mathbf{m}_i, \mathbf{P}_i) \quad (2.10)$$

the goal is to find the parameters  $\alpha_i, \mathbf{m}_i, \mathbf{P}_i$ . If the GMM approximation is required to lay along any chosen eigenvector of  $\mathbf{P}$ , such as it is illustrated in Fig. 2.1 for the largest eigenvector, the Jordan decomposition is to be accomplished,

$$\mathbf{P} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T, \quad \mathbf{V} = [\mathbf{v}_1, \dots, \mathbf{v}_n], \quad \mathbf{\Lambda} = \text{diag}\{\lambda_1, \dots, \lambda_n\} \quad (2.11)$$

where  $\lambda_i, \mathbf{v}_i \forall i$  are the eigenvalues and eigenvectors, respectively. Then, the parameters can be computed using

$$\alpha_i = \tilde{\alpha}_i \alpha, \quad \mathbf{m}_i = \mathbf{m} + \sqrt{\lambda_k} \tilde{m}_i \mathbf{v}_k, \quad \mathbf{P}_i = \mathbf{V} \tilde{\mathbf{\Lambda}}_i \mathbf{V}^T \quad (2.12)$$

$$\tilde{\mathbf{\Lambda}}_i = \text{diag}\{\lambda_1, \dots, \tilde{\sigma}_i^2 \lambda_k, \dots, \lambda_n\} \quad (2.13)$$



where  $k$  establishes the chosen eigenvector along which the Gaussian PDF is to be split. If the largest one is to be chosen, then  $k = \arg_i \max\{\lambda_i\}$ .

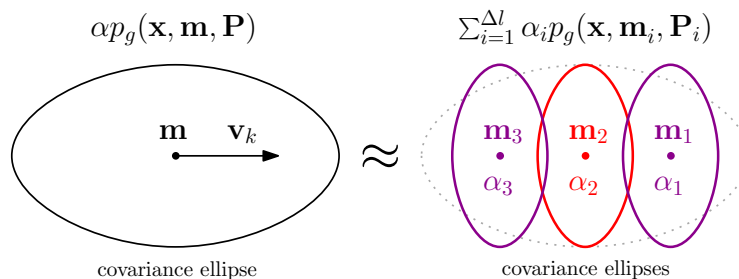


Figure 2.1: Illustration of splitting a Gaussian distribution.

Table 2.1: Three-component splitting library

$i$	$\tilde{\alpha}_i$	$\tilde{m}_i$	$\tilde{\sigma}_i$
1	0.2252246249	-1.0575154615	0.6715662887
2	0.5495507502	0	0.6715662887
3	0.2252246249	1.0575154615	0.6715662887

Table 2.2: Five-component splitting library

$i$	$\tilde{\alpha}_i$	$\tilde{m}_i$	$\tilde{\sigma}_i$
1	0.0763216491	-1.6899729111	0.6715662887
2	0.2474417860	-0.8009283834	0.6715662887
3	0.35247313	0	0.6715662887
4	0.2474417860	0.8009283834	0.4422555386
5	0.076321649	1.6899729111	0.4422555386

## 2.3 Differential Entropy of a Gaussian Distribution

The method AEGIS, explained in Section 3.7, uses entropy in order to indicate when to start the splitting process discussed in the section above. Therefore, entropy of the nonlinear and linearized system state is to be discussed in this section.

A Differential (Shannon) entropy of a random variable  $\mathbf{x} \in \mathbb{R}^n$  with PDF  $p(\mathbf{x})$  is given by [8]

$$H(\mathbf{x}) = \int_{\mathcal{S}} p(\mathbf{x}) \log p(\mathbf{x}) d\mathbf{x} = E[-\log p(\mathbf{x})] \quad (2.14)$$

where  $\mathcal{S} = \{\mathbf{x} : p(\mathbf{x}) > 0\}$  is a support set of the PDF. If the PDF is Gaussian, the differential entropy can be written in terms of covariance matrix,

$$\begin{aligned} H(\mathbf{x}) &= E \left[ -\log \frac{1}{|2\pi\mathbf{P}|^{1/2}} - \log \exp\left\{-\frac{1}{2}(\mathbf{x} - \mathbf{m})^T \mathbf{P}^{-1}(\mathbf{x} - \mathbf{m})\right\} \right] \\ &= E \left[ \frac{1}{2} \log |2\pi\mathbf{P}| + \frac{1}{2}(\mathbf{x} - \mathbf{m})^T \mathbf{P}^{-1}(\mathbf{x} - \mathbf{m}) \right] \\ &= \frac{1}{2} \log |2\pi\mathbf{P}| + \frac{1}{2} E[(\mathbf{x} - \mathbf{m})^T \mathbf{P}^{-1}(\mathbf{x} - \mathbf{m})] \\ &= \frac{1}{2} \log |2\pi\mathbf{P}| + \frac{1}{2} \text{trace}\{E[(\mathbf{x} - \mathbf{m})(\mathbf{x} - \mathbf{m})^T \mathbf{P}^{-1}]\} \\ &= \frac{1}{2} \log |2\pi\mathbf{P}| + \frac{1}{2} \text{trace}\{\mathbf{P}\mathbf{P}^{-1}\} = \frac{1}{2} \log |2\pi\mathbf{P}| + \frac{1}{2} \text{trace}\{\mathbf{I}\} \\ &= \frac{1}{2} \log |2\pi\mathbf{P}| + \frac{n}{2} = \frac{1}{2} \log |2\pi\mathbf{P}| + \frac{1}{2} \log e^n = \frac{1}{2} \log |2\pi\mathbf{P}|e^n = \frac{1}{2} \log |2\pi e\mathbf{P}| \quad (2.15) \end{aligned}$$

where  $\mathbf{I} \in \mathbb{R}^{n \times n}$  denotes the identity matrix. The differential entropy describes the lack of information about the random variable.

For the Gaussian distribution, the differential entropy drawback is that if  $|\mathbf{P}| < \frac{1}{(2\pi e)^n}$ , the entropy becomes negative, which happens if  $p_g(\mathbf{m}) > e^{n/2}$ , where  $\mathbf{m}$  is the mean.

### 2.3.1 Entropy of a Linearized System State

Assuming the initial state has a Gaussian PDF, its entropy is defined by Eq. (2.15), which temporal derivative is

$$\dot{H}(\mathbf{x}(t)) = \frac{d}{dt} \left( \frac{1}{2} \log |2\pi e \mathbf{P}(t)| \right) \quad (2.16)$$

$$= \frac{1}{2} \frac{1}{|2\pi e \mathbf{P}(t)|} \frac{d}{dt} |2\pi e \mathbf{P}(t)| \text{trace} \left\{ \frac{1}{2\pi e} \mathbf{P}^{-1}(t) \frac{d}{dt} (2\pi e \dot{\mathbf{P}}(t)) \right\} \quad (2.17)$$

$$= \frac{1}{2} \text{trace} \{ \mathbf{P}^{-1}(t) \dot{\mathbf{P}}(t) \} \quad (2.18)$$

with initial condition defined by  $\mathbf{P}(t_0)$ .

Assume a linear system of the form

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{m}(t), t) + \mathbf{A}(\mathbf{m}(t), t)[\mathbf{x}(t) - \mathbf{m}(t)], \quad \mathbf{A}(\mathbf{m}(t), t) = \left. \frac{\partial \mathbf{f}(\mathbf{x}, t)}{\partial \mathbf{x}} \right|_{\mathbf{x}=\mathbf{m}(t)} \quad (2.19)$$

where  $\mathbf{m}(t)$  is the mean. For such system, covariance  $\mathbf{P}(t)$  time evolution is governed by

$$\dot{\mathbf{P}}(t) = \mathbf{A}(\mathbf{m}(t), t)\mathbf{P}(t) + \mathbf{P}(t)\mathbf{A}^T(\mathbf{m}(t), t), \quad \mathbf{P}(0) = \mathbf{P}_0 \quad (2.20)$$

See Section 3.4.2 for derivation.

Therefore the entropy of a linearized system is governed by

$$\dot{H}(\mathbf{x}(t)) = \frac{1}{2} \text{trace} \{ \mathbf{P}^{-1} \mathbf{A} \mathbf{P} + \mathbf{P}^{-1} \mathbf{P} \mathbf{A}^T \} = \frac{1}{2} \text{trace} \{ \mathbf{A} \mathbf{P} \mathbf{P}^{-1} \} + \frac{1}{2} \text{trace} \{ \mathbf{A}^T \} \quad (2.21)$$

$$= \text{trace} \{ \mathbf{A}(\mathbf{m}(t), t) \} \quad (2.22)$$

with initial condition defined by  $\mathbf{m}(t_0)$ . Such entropy evolution can be computed for both LinCov (see Section 3.3) and FOTE based (see Section 3.4) propagation methods.

---

# Uncertainty Propagators

In this chapter, uncertainty propagation methods are theoretically stated and analyzed. The selected methods include Monte Carlo (MC) simulation, local linearization (LinCov and the first order Taylor expansion based propagation - FOTE) and statistical linearization methods (CADET), unscented transformation (UT), and entropy based method for adaptive GMM update (AEGIS). The descriptions are briefly provided with derivations, a graphical illustration<sup>1</sup>, and a theoretical analysis (pros and cons, discussion, etc.).

Methods described herein are certainly not the only ones that exist. The current state of art is summarized in Fig. 3.1, as it was described in [3], methods that are analyzed within this thesis are highlighted in **red**.

The methods can be divided into few categories which are linear, nonlinear and other methods.

Linear methods focus on a linear assumption about the system model description (local linearization - discrete-time LinCov, and continuous-time FOTE based propagation) and uncertainty transformation (statistical linearization - CADET).

Nonlinear methods take the nonlinear nature of the system model into consideration. Sample-based methods focus on the system state behavior (state propagation through the transformation) - Monte Carlo simulation, unscented transformation and polynomial chaos expansion. Dynamics-based methods focus on the transformation and its approximation itself (e.g. higher-order Taylor expansion) - state transition tensors and differential algebra technique. PDF-based methods focus on the approximation of the PDF (the uncertainty representation) - GMM and solving FPE.

---

<sup>1</sup>Graphical illustrations are taken from the Low Earth Orbit (LEO) test case (position results of a single orbit of the space object) discussed in Section 4.2. The key element of each method is highlighted in **red**.

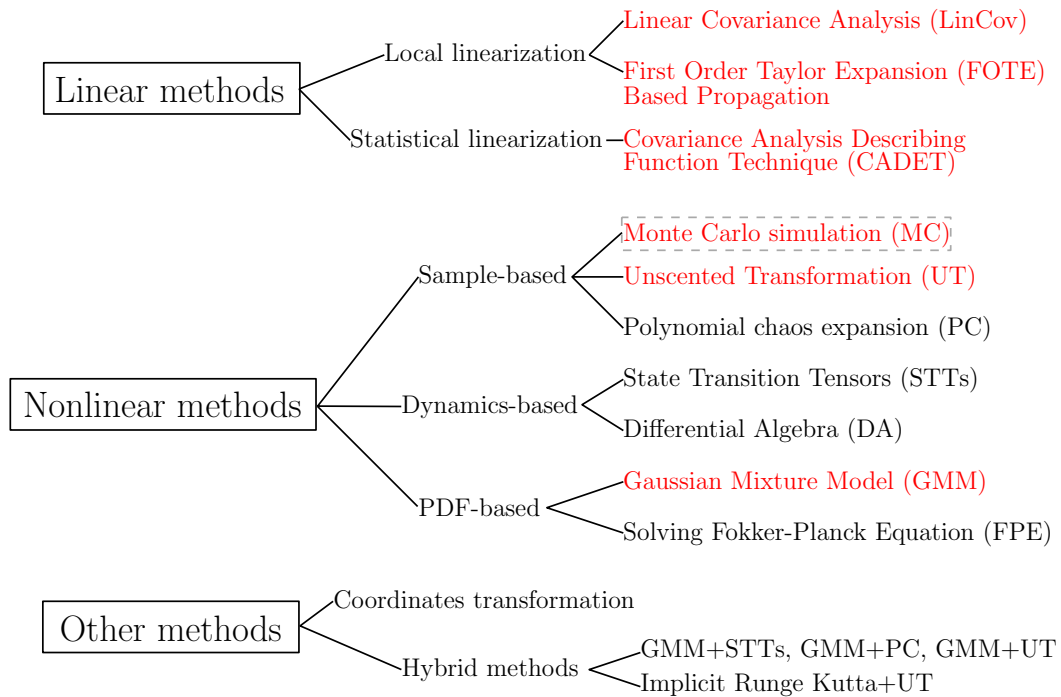


Figure 3.1: Uncertainty propagation methods outline.

The other methods try to solve the problem in more convenient sense (coordinates transformation) or try to make combination of different approaches.

The main objectives using which the methods are compared, are computational costs, accuracy, system suitability and the overall results.

Some of the methods suffer from the curse of dimensionality, meaning that the desired accuracy can be reached only if dimensions of some parameters increase without bounds. A goal is to find a method with the best ratio of accuracy to speed.

Some of the methods require that the system dynamics is differentiable (the derivative  $\mathbf{A}(\mathbf{x}(t), t)$  exists), some do not.

### 3.1 Problem Statement

Consider the following nonlinear system with an uncertain initial condition, as a default theoretical model in this thesis,

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), t), \quad \mathbf{x}(t_0) = \mathbf{x}_0 \tag{3.1}$$

where  $\mathbf{x}(t) \in \mathbb{R}^n$  is a state vector, dynamics  $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^n$  is a sufficiently differentiable function, and  $\mathbf{x}_0$  is assumed to be a random variable with a PDF  $p(\mathbf{x}_0)$ . Mostly, the initial PDF is taken to be Gaussian  $p_g(\mathbf{x}_0; \mathbf{m}_0, \mathbf{P}_0)$ . Because the subject of this thesis is to construct a state uncertainty prediction, consider there is only a single initial measurement which determines  $\mathbf{m}_0$  and  $\mathbf{P}_0$ , and no other measurements for  $t > 0$  are available. Also, no input signals such as actuation, control, disturbance or noise are considered. The objective then is to find a plausible statistical properties, such as mean  $\mathbf{m}$  and covariance matrix  $\mathbf{P}$  of  $\mathbf{x}(t)$  at any desired time  $t$ .

A graphical illustration of this process is shown in Fig. 3.2, where the propagation process is split into a branch which represents the real uncertainty propagation (which satisfies the Fokker-Planck equation and results into the true density, see Section 3.1.1) and a branch which represents the approximation procedure (use of an uncertainty propagation method) which may conditionally converge to the true PDF.

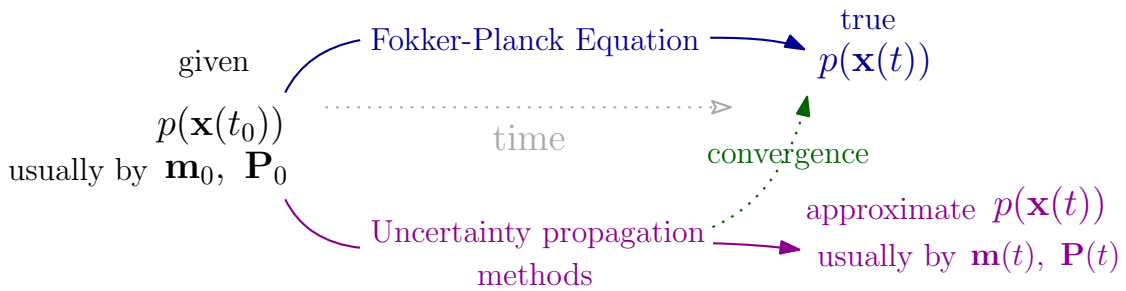


Figure 3.2: Illustration of uncertainty propagation.

Note that if the initial condition is not random, the system state at any time can be implicitly denoted as a numerical solution to the differential equation 3.1,

$$\mathbf{x}(t) = \phi(\mathbf{x}_0, t_0, t) \quad (3.2)$$

where  $\phi(\mathbf{x}_0, t_0, t)$  is the state solution flow.

### 3.1.1 Fokker-Planck Equation

The exact time evolution of the PDF  $p(\mathbf{x}(t), t)$  is given by the Fokker-Planck equation (FPE) in terms of no diffusion [9]:

$$\frac{\partial}{\partial t} p(\mathbf{x}(t), t) = - \sum_{i=1}^n \frac{\partial}{\partial x_i(t)} [p(\mathbf{x}(t), t) \cdot f_i(\mathbf{x}(t), t)] \quad (3.3)$$

where the initial PDF is taken to be  $p_g(\mathbf{x}_0; \mathbf{m}_0, \mathbf{P}_0)$ . As the PDF time evolution depends on the system dynamics  $\mathbf{f}(\mathbf{x}(t), t)$ , the initial Gaussian distribution can easily become non-Gaussian. As studied in [1] or [2], the used coordinate system has big impact on this problem. Obtaining an exact solution of Eq. (3.3) is possible only in special cases such as linear systems, e.g. [10].

Note that the model settled in Eq. (3.1) can be further generalized in terms of state diffusion as it is studied e.g. in the paper [3], but this model will not be under consideration in this thesis.

## 3.2 Monte Carlo Simulation

Monte Carlo (MC) simulation is a technique based on random number generation that captures the shape of the transformed PDF by a set of samples.

### 3.2.1 The Method Description

First, a set of samples  $\{\mathbf{x}_{i,0}\}_{i=1}^N$  derived from a given initial distribution  $p(\mathbf{x}_0)$  is generated. If the initial distribution is taken to be Gaussian, each sample can be generated with a Gaussian pseudorandom number generator. Then, each sample  $\mathbf{x}_{i,0}$  is transformed through the nonlinear dynamics from  $t_0$  to  $t$ , which can be denoted using the state solution flow function as

$$\mathbf{x}_i(t) = \phi(\mathbf{x}_{i,0}, t_0, t), \quad i = 1, \dots, N \quad (3.4)$$

as a result of a numerical integration. Therefore, the first two moments can be computed via equations for sample mean and sample variance,

$$\mathbf{m}(t) = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i(t) \quad (3.5)$$

$$\mathbf{P}(t) = \frac{1}{N-1} \sum_{i=1}^N [\mathbf{x}_i(t) - \mathbf{m}(t)][\mathbf{x}_i(t) - \mathbf{m}(t)]^T \quad (3.6)$$

### 3.2.2 Analysis and Discussion

The results<sup>2</sup> of the MC simulation approach to the true distribution when  $N \rightarrow +\infty$ , therefore it is considered reliable to validate other uncertainty propagation methods [3]. It is suitable for any nonlinear system as it captures a shape of an arbitrarily transformed initial PDF. The results are very intuitive and easy to understand. Also, the algorithm is easy to implement.

---

<sup>2</sup>The transformed MC samples can be used to create for example a Dirac mixture model distribution (DMM) [11], or a histogram PDF.



The biggest disadvantage of this method is that to achieve high accuracy,  $N$  has to be high enough, which makes the method computationally expensive.

A graphical illustration is shown in Fig. 3.3.

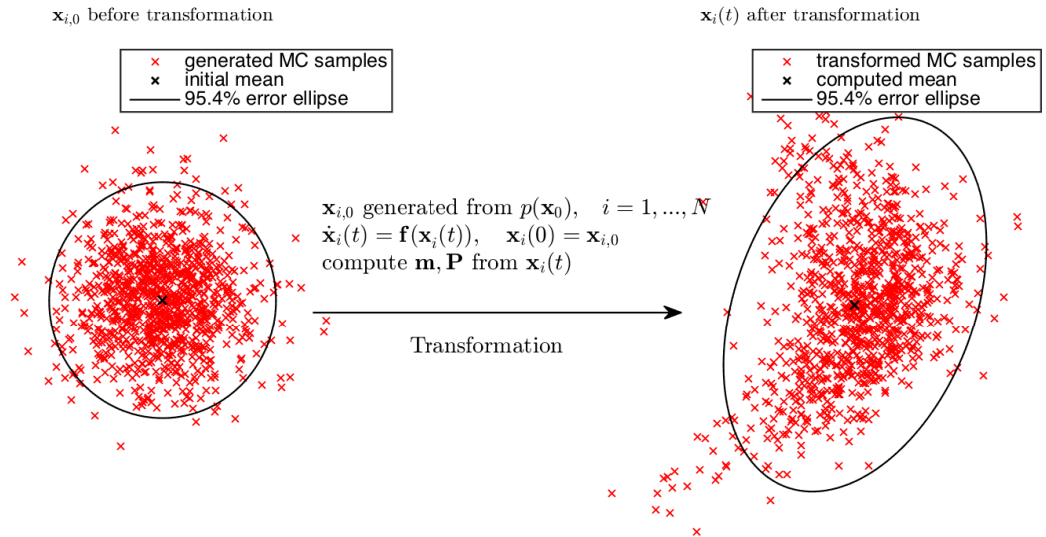


Figure 3.3: Monte Carlo simulation graphical illustration, with 1000 MC samples.

### 3.3 LinCov

“Linear Covariance” (LinCov) method is a discrete-time propagation method based on a local linearization along a reference trajectory  $\bar{\mathbf{x}}(t)$ . The goal is to approximate the covariance matrix. A general formulation of LinCov can be found in [12].

#### 3.3.1 The Method Derivation and Description

A local linearization of  $\mathbf{f}(\mathbf{x}(t), t)$  along a reference trajectory  $\bar{\mathbf{x}}(t)$  is to be created using the first-order Taylor expansion,

$$\dot{\mathbf{x}}(t) \approx \mathbf{f}(\bar{\mathbf{x}}(t), t) + \mathbf{A}(\bar{\mathbf{x}}(t), t)\mathbf{e}(t) \quad (3.7)$$

$$\mathbf{e}(t) = \mathbf{x}(t) - \bar{\mathbf{x}}(t), \quad \mathbf{A}(\bar{\mathbf{x}}(t), t) = \left. \frac{\partial \mathbf{f}(\mathbf{x}, t)}{\partial \mathbf{x}} \right|_{\mathbf{x}=\bar{\mathbf{x}}(t)} \quad (3.8)$$

where  $\mathbf{e}(t)$  is the state error vector. The reference trajectory  $\bar{\mathbf{x}}(t)$  might be a solution to the differential equation (3.1) with an initial condition given by the initial mean  $\mathbf{m}_0$ . Because  $\mathbf{f}(\bar{\mathbf{x}}(t), t) = \dot{\bar{\mathbf{x}}}(t)$ , the Eq. (3.7) yields,

$$\dot{\mathbf{e}}(t) \approx \mathbf{A}(\bar{\mathbf{x}}(t), t)\mathbf{e}(t) \quad (3.9)$$

Assuming the state error vector retains small, the analytic solution can be expressed as

$$\mathbf{e}(t) = \Phi(\bar{\mathbf{x}}(t), \Delta t)\mathbf{e}(t_0), \quad \Phi(\bar{\mathbf{x}}(t), \Delta t) \triangleq e^{\mathbf{A}(\bar{\mathbf{x}}(t), t) \cdot \Delta t} \quad (3.10)$$

where  $\Phi(\bar{\mathbf{x}}(t), \Delta t)$  is the state transition matrix (STM). Note that  $t = t_0 + \Delta t$ . The state solution flow equation for this linearized model is

$$\mathbf{x}(t) = \bar{\mathbf{x}}(t) + \Phi(\bar{\mathbf{x}}(t), \Delta t)\mathbf{x}(t_0) - \Phi(\bar{\mathbf{x}}(t), \Delta t)\bar{\mathbf{x}}(t_0) \quad (3.11)$$

Assuming  $E[\mathbf{x}(t_0)] = \bar{\mathbf{x}}(t_0)$ , the mean  $\mathbf{m}(t) = E[\mathbf{x}(t)]$  retains simply the reference trajectory. The propagation of the covariance matrix is given by

$$\mathbf{P}(t) = E[(\mathbf{x}(t) - \mathbf{m}(t))(\dots)^T] \quad (3.12)$$

$$= E[\Phi(\bar{\mathbf{x}}(t), \Delta t)(\mathbf{x}(t_0) - \mathbf{m}(t_0))(\dots)^T] \quad (3.13)$$

$$\Rightarrow \mathbf{P}(t) = \Phi(\bar{\mathbf{x}}(t), \Delta t)\mathbf{P}(t_0)\Phi^T(\bar{\mathbf{x}}(t), \Delta t) \quad (3.14)$$

which is the main result of the LinCov method.

### 3.3.2 Analysis and Discussion

Unlike the Monte Carlo simulation, LinCov needs to solve the object trajectory only once (to find the reference trajectory  $\bar{\mathbf{x}}(t)$ ), which makes it way more computationally effective.

A disadvantage is that Eq. (3.14) holds only if  $\Delta t \rightarrow 0$  due to the local linearization effect. This fact makes this method iterative, time discretization  $\Delta t$  (which becomes a design parameter) should be small, otherwise the results will be less accurate. This makes LinCov inaccurate for strongly nonlinear systems. Moreover, in order to find the matrix function  $\mathbf{A}(\mathbf{x}(t), t)$ , the derivative of the system dynamics must exist.

Note that the LinCov propagation scheme described in Eq. (3.14), also appears in a discrete-time extended Kalman filter. A graphical illustration of the LinCov method is shown in Fig. 3.4.

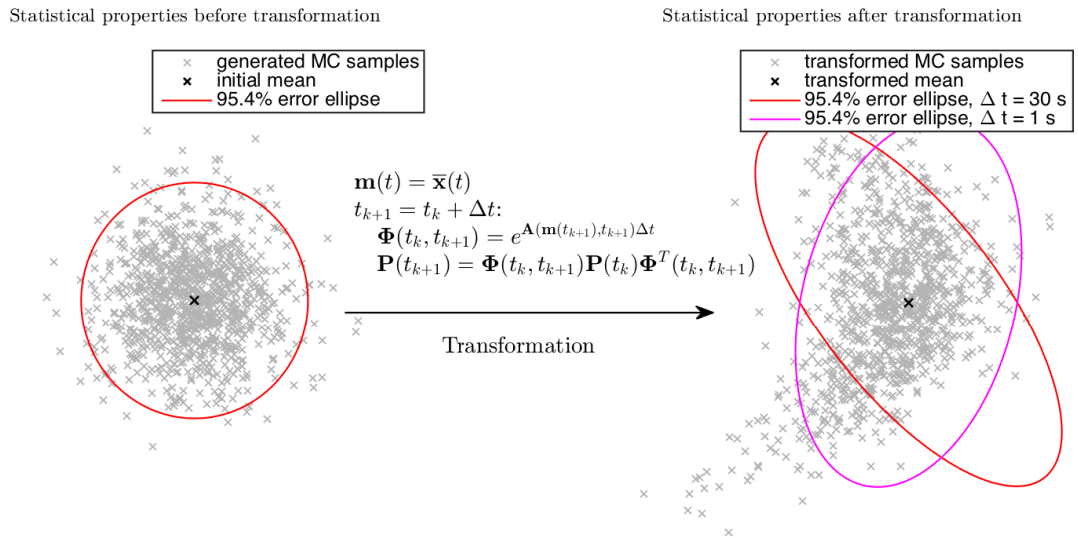


Figure 3.4: LinCov technique graphical illustration with comparison to 1000 Monte Carlo samples (gray).

## 3.4 FOTE Based Propagation

The first Order Taylor Expansion (FOTE) based propagation is a continuous-time propagation method based on a local linearization along an approximated mean  $\mathbf{m}(t)$ .

### 3.4.1 The Method Description

Using Eq. (3.7) a local linearization of  $\mathbf{f}(\mathbf{x}(t), t)$  along the mean  $\mathbf{m}(t)$  is given by a first order Taylor series expansion:

$$\dot{\mathbf{x}}(t) \approx \mathbf{f}(\mathbf{m}(t), t) + \mathbf{A}(\mathbf{m}(t), t)[\mathbf{x}(t) - \mathbf{m}(t)] \quad (3.15)$$

where  $\mathbf{A}(\mathbf{m}(t), t)$  is the first order derivative, see Eq. (3.8). Then the propagation governing equations are

$$\dot{\mathbf{m}}(t) = \mathbf{f}(\mathbf{m}(t), t), \quad \mathbf{m}(0) = \mathbf{m}_0 \quad (3.16)$$

$$\dot{\mathbf{P}}(t) = \mathbf{A}(\mathbf{m}(t), t)\mathbf{P}(t) + \mathbf{P}(t)\mathbf{A}^T(\mathbf{m}(t), t), \quad \mathbf{P}(0) = \mathbf{P}_0 \quad (3.17)$$

Eq. (3.17) formally is the continuous-time Lyapunov Equation. In terms of our objective, it can be derived for example from the FPE [10]. An alternative derivation is shown as follows.

### 3.4.2 The Alternative Derivation

For convenience, the notation  $\mathbf{A} = \mathbf{A}(\mathbf{m}(t), t)$  will be used. Let  $\mathbf{r}(t) = \mathbf{x}(t) - \mathbf{m}(t)$ , then

$$\dot{\mathbf{r}}(t) = \dot{\mathbf{x}}(t) - \dot{\mathbf{m}}(t) = \cancel{\mathbf{f}(\mathbf{m}(t), t)} + \mathbf{A}[\mathbf{x}(t) - \mathbf{m}(t)] - \cancel{\mathbf{f}(\mathbf{m}(t), t)} = \mathbf{A}\mathbf{r}(t) \quad (3.18)$$

which can be approximated by Euler's rule:

$$\mathbf{r}(t + \Delta t) = \mathbf{r}(t) + \mathbf{A}\mathbf{r}(t)\Delta t \quad (3.19)$$

Because  $\mathbf{P}(t) = E[\mathbf{r}(t)\mathbf{r}^T(t)]$ , we get

$$E[\mathbf{r}(t + \Delta t)\mathbf{r}^T(t + \Delta t)] = E[(\mathbf{r}(t) + \mathbf{A}\mathbf{r}(t)\Delta t)(\dots)^T] \quad (3.20)$$

$$\mathbf{P}(t + \Delta t) = \mathbf{P}(t) + \mathbf{A}\mathbf{P}(t)\Delta t + \Delta t\mathbf{P}(t)\mathbf{A}^T + \mathbf{A}\mathbf{P}(t)\mathbf{A}^T\Delta^2 t \quad (3.21)$$

$$\frac{\mathbf{P}(t + \Delta t) - \mathbf{P}(t)}{\Delta t} = \mathbf{A}\mathbf{P}(t) + \mathbf{P}(t)\mathbf{A}^T + \mathbf{A}\mathbf{P}(t)\mathbf{A}^T\Delta t \quad (3.22)$$

Taking the limit as  $\Delta t \rightarrow 0$  results into the desired equation:

$$\lim_{\Delta t \rightarrow 0} \frac{\mathbf{P}(t + \Delta t) - \mathbf{P}(t)}{\Delta t} = \dot{\mathbf{P}}(t) = \mathbf{A}(\mathbf{m}(t), t)\mathbf{P}(t) + \mathbf{P}(t)\mathbf{A}^T(\mathbf{m}(t), t) \quad (3.23)$$

### 3.4.3 Analysis and Discussion

This linear propagation scheme is widely used because of its good ratio of accuracy and speed. The object trajectory has to be solved only once to find the mean. No design parameter is needed.

The accuracy is sabotaged by local linearization effects. Like in the LinCov analysis, in order to find the matrix function  $\mathbf{A}(\mathbf{m}(t), t)$ , derivative of the system dynamics must exist.

Note that LinCov and FOTE based propagation are both linear propagators (both are de facto first order Taylor expansion based). The FOTE based propagation can be used only in terms of the continuous-time, while LinCov can be readily used both in terms of continuous-time and discrete-time sense but with respect to the given discretization  $\Delta t$ . Note that the FOTE based propagation scheme also appears in the continuous-time extended Kalman filter. A graphical illustration of the FOTE based propagation is shown in Fig. 3.5.

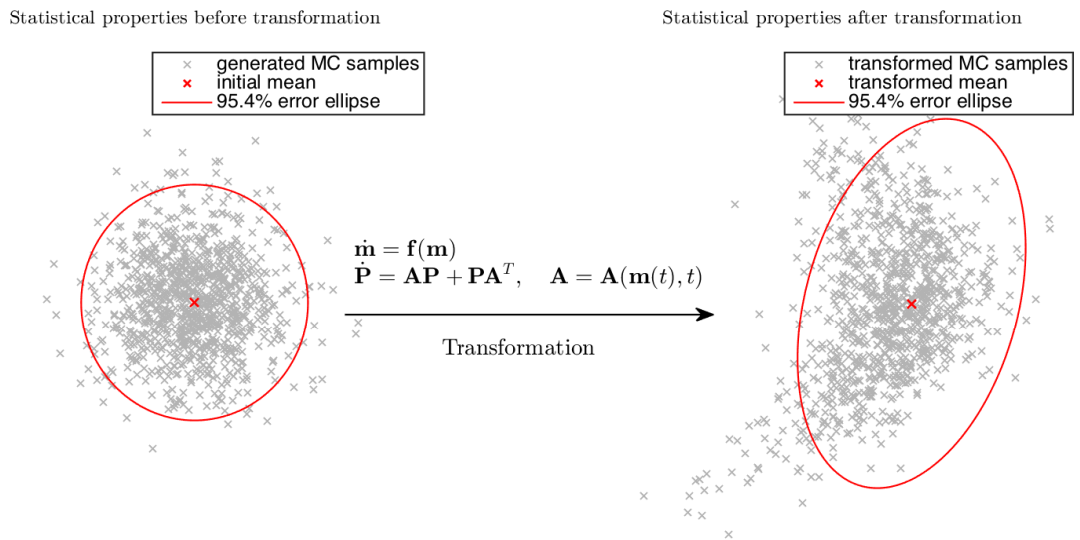


Figure 3.5: FOTE based propagation graphical illustration with comparison to 1000 Monte Carlo samples (gray).

## 3.5 CADET

“The Covariance Analysis Describing Function Technique” (CADET) is a method that uses a statistical linearization. A formulation of this method can be found in [13, 3].

### 3.5.1 The Method Derivation and Description

A statistical linearization is to approximate the state vector  $\mathbf{x}(t)$  and the leading function  $\mathbf{f}(\mathbf{x}(t), t)$  with a linear expression

$$\mathbf{x}(t) = \mathbf{m}(t) + \mathbf{r}(t) \quad (3.24)$$

$$\widehat{\mathbf{f}}(\mathbf{x}(t), t) = \mathbf{N}_m(t)\mathbf{m}(t) + \mathbf{N}_r(t)\mathbf{r}(t) \quad (3.25)$$

where  $\mathbf{m}(t) = E[\mathbf{x}(t)]$  and  $\mathbf{r}(t)$  is a zero-mean independent random process representing state deviations,  $\mathbf{P}(t) = E[\mathbf{r}(t)\mathbf{r}^T(t)]$ . Note that  $E[\mathbf{r}(t)\mathbf{m}^T(t)] = E[\mathbf{m}(t)\mathbf{r}^T(t)] = \mathbf{0}$ .  $\mathbf{N}_m(t)$  and  $\mathbf{N}_r(t)$  are the so-called multiple-input describing function gain matrices that are chosen to minimize the mean square approximation error  $\mathbf{e}_f(t)$ ,

$$\mathbf{e}_f(t) = \mathbf{f}(\mathbf{x}(t), t) - \widehat{\mathbf{f}}(\mathbf{x}(t), t) \quad (3.26)$$

$$= \mathbf{f}(\mathbf{x}(t), t) - \mathbf{N}_m(t)\mathbf{m}(t) - \mathbf{N}_r(t)\mathbf{r}(t) \quad (3.27)$$

For convenience, the arguments of the functions will be omitted. Then, the criterion to minimize is

$$J = E[\mathbf{e}_f^T \mathbf{S} \mathbf{e}_f] = \text{trace } E[\mathbf{e}_f \mathbf{e}_f^T] \mathbf{S} \quad (3.28)$$

$$= E[\mathbf{f}^T \mathbf{S} \mathbf{f} - 2\mathbf{f}^T \mathbf{S} \mathbf{N}_m \mathbf{m} - 2\mathbf{f}^T \mathbf{S} \mathbf{N}_r \mathbf{r} + 2\mathbf{m}^T \mathbf{N}_m^T \mathbf{S} \mathbf{N}_r \mathbf{r} + \mathbf{m}^T \mathbf{N}_m^T \mathbf{S} \mathbf{N}_m \mathbf{m} + \mathbf{r}^T \mathbf{N}_r^T \mathbf{S} \mathbf{N}_r \mathbf{r}] \quad (3.29)$$

where  $\mathbf{S}$  is a symmetric positive semi-definite matrix. Minimization of  $J$  leads to

$$\frac{\partial}{\partial \mathbf{N}_m} J = -2E[\mathbf{f}^T] \mathbf{S} \mathbf{m} + 2\mathbf{m}^T \mathbf{N}_m^T \mathbf{S} \mathbf{m} + \underline{2\mathbf{m}^T \mathbf{S} \mathbf{N}_r} E[\mathbf{r}] \stackrel{!}{=} 0 \quad (3.30)$$

$$\frac{\partial}{\partial \mathbf{N}_r} J = -2E[\mathbf{f}^T \mathbf{S} \mathbf{r}] + \underline{2\mathbf{m}^T \mathbf{N}_m^T \mathbf{S} E[\mathbf{r}]} + 2E[\mathbf{r}^T \mathbf{S} \mathbf{N}_r \mathbf{r}] \stackrel{!}{=} 0 \quad (3.31)$$

because  $E[\mathbf{r}] = \mathbf{0}$ . Taking Eq. (3.30),  $\mathbf{N}_m$  can be expressed as

$$2\mathbf{m}^T \mathbf{N}_m^T \mathcal{S} \mathbf{m} = 2E[\mathbf{f}^T] \mathcal{S} \mathbf{m} \quad /()^T \quad (3.32)$$

$$\mathbf{m}^T \mathbf{N}_m^T \mathbf{m} = \mathbf{m}^T E[\mathbf{f}] \quad (3.33)$$

$$\Rightarrow \mathbf{N}_m(t) \mathbf{m}(t) = E[\mathbf{f}(\mathbf{x}(t), t)] \quad (3.34)$$

Taking Eq. (3.31),  $\mathbf{N}_r$  can be expressed as

$$E[\mathbf{r}^T \mathcal{S} \mathbf{N}_r \mathbf{r}] = E[\mathbf{f} \mathcal{S} \mathbf{r}] \quad (3.35)$$

$$\text{trace } E[\mathbf{r} \mathbf{r}^T] \mathbf{N}_r = \text{trace } E[\mathbf{f} \mathbf{r}^T] \quad (3.36)$$

$$\Rightarrow \mathbf{N}_r(t) = E[\mathbf{f}(\mathbf{x}(t), t) \mathbf{r}^T(t)] \mathbf{P}^{-1}(t) \quad (3.37)$$

Therefore, the statistical linearization [combining Eq. (3.1) and (3.25)] is in the form

$$\dot{\mathbf{x}}(t) = \widehat{\mathbf{f}}(\mathbf{x}(t), t) = \underbrace{E[\mathbf{f}(\mathbf{x}(t), t)]}_{\mathbf{N}_m \mathbf{m}} + \underbrace{E[\mathbf{f}(\mathbf{x}(t), t) \mathbf{r}^T(t)] \mathbf{P}^{-1}(t)}_{\mathbf{N}_r} \mathbf{r}(t) \quad (3.38)$$

Moreover, under the assumption of  $\mathbf{x}(t)$  being Gaussian,  $\mathbf{N}_r(t)$  can be computed [13] using expression

$$\mathbf{N}_r(t) = \frac{\partial}{\partial \mathbf{m}(t)} E[\mathbf{f}(\mathbf{x}(t), t)] \quad (3.39)$$

which might be approximated by  $\mathbf{N}_r(t) \approx \mathbf{A}(\mathbf{m}(t), t)$ . The mean propagation is

$$\dot{\mathbf{m}}(t) = E[\widehat{\mathbf{f}}(\mathbf{x}(t), t)] = \mathbf{N}_m(\mathbf{m}, \mathbf{P}, t) \mathbf{m}(t) = E[\mathbf{f}(\mathbf{x}(t), t)] \quad (3.40)$$

Taking Eq. (2.5) the covariance propagation is as follows

$$\dot{\mathbf{P}}(t) = E[\widehat{\mathbf{f}}(\mathbf{m}^T + \mathbf{r}^T) + (\mathbf{m} + \mathbf{r}) \widehat{\mathbf{f}}^T] - E[\widehat{\mathbf{f}}] \mathbf{m}^T - \mathbf{m} E[\widehat{\mathbf{f}}^T] = E[\widehat{\mathbf{f}} \mathbf{r}^T] + E[\mathbf{r} \widehat{\mathbf{f}}] \quad (3.41)$$

$$= E[(\mathbf{N}_m \mathbf{m} + \mathbf{N}_r \mathbf{r}) \mathbf{r}^T] + E[\mathbf{r} (\mathbf{m}^T \mathbf{N}_m^T + \mathbf{r}^T \mathbf{N}_r^T)] = E[\mathbf{N}_r \mathbf{r} \mathbf{r}^T] + E[\mathbf{r} \mathbf{r}^T \mathbf{N}_r] \quad (3.42)$$

$$= \mathbf{N}_r(\mathbf{m}, \mathbf{P}, t) \mathbf{P}(t) + \mathbf{P}(t) \mathbf{N}_r^T(\mathbf{m}, \mathbf{P}, t) \quad (3.43)$$

$$= E[\mathbf{f}(\mathbf{x}(t), t) \mathbf{r}^T(t)] + E[\mathbf{r}(t) \mathbf{f}^T(\mathbf{x}(t), t)] \quad (3.44)$$



### 3.5.2 Analysis and Discussion

A benefit of this method is its suitability to high-order systems, whose derivatives do not exist, systems with multiple nonlinearities, inputs or nongaussian statistical description.

The implementation might be tricky due to the need to calculate  $E[\mathbf{f}(\mathbf{x}(t), t)]$  and  $E[\mathbf{f}(\mathbf{x}(t), t)\mathbf{r}^T(t)]$  which can be high-variate integrals with the need to evaluate current  $p(\mathbf{x}(t))$ . That might be done with some approximation e.g. some quadrature rule (UT, see Section 3.6) or a sparse grid integration [14]. However, this can make the results of CADET inaccurate for nonlinear systems.

The CADET method can be used for example in missile guidance systems [13], or estimation. A graphical illustration of the CADET method is shown in Fig. 3.6.

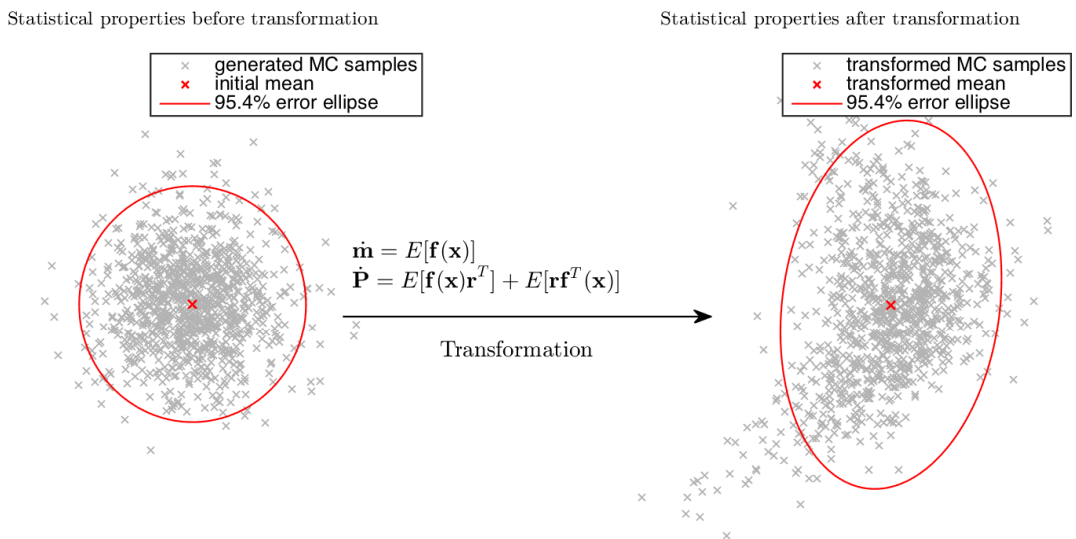


Figure 3.6: CADET method graphical illustration with comparison to 1000 Monte Carlo samples (gray).

## 3.6 UT

The continuous-time unscented transformation [15] is a nonlinear propagation method that uses a set of so-called weighted sigma-points generated from the initial distribution that are subsequently transformed through the nonlinear function to yield the transformed mean and covariance matrix. Unlike the Monte Carlo samples, the sigma-points are deterministically chosen from the initial distribution, so that they capture a specific information about it. Various ways to determine the sigma-points exist [17], the symmetric sigma-point set is presented herein.

### 3.6.1 The Method Description

Let  $n$  be the dimensionality of the state vector  $\mathbf{x}$ . In order to determine the sigma-points, the square-root factor  $\mathbf{S}$  of the initial covariance matrix has to be found,

$$\mathbf{P}_0 = \mathbf{S}\mathbf{S}^T, \quad \mathbf{S} = [\mathbf{s}_1, \dots, \mathbf{s}_n] \quad (3.45)$$

which can be accomplished for example via the Cholesky factorization, or via a singular-value decomposition. Then, the symmetric sigma-point set  $\{\mathcal{X}_{i,0}\}_{i=1}^{2n}$  is determined as

$$\mathcal{X}_{i,0} = \mathbf{m}_0 + \sqrt{n}\mathbf{s}_i, \quad i = 1, \dots, n \quad (3.46)$$

$$\mathcal{X}_{i,0} = \mathbf{m}_0 - \sqrt{n}\mathbf{s}_i, \quad i = n+1, \dots, 2n \quad (3.47)$$

It can be shown [15], that this sample set has the mean and covariance of the initial distribution  $p(\mathbf{x}_0)$ . Then, each sigma-point is transformed through the nonlinear transformation,

$$\mathcal{X}_i(t) = \phi(\mathcal{X}_{i,0}, t_0, t), \quad i = 1, \dots, 2n \quad (3.48)$$

Then, the first two moments of  $\mathbf{x}(t)$  can be computed via equations

$$\mathbf{m}(t) = \sum_{i=1}^{2n} w_i \mathcal{X}_i(t) \quad (3.49)$$

$$\mathbf{P}(t) = \sum_{i=1}^{2n} w_i [\mathcal{X}_i(t) - \mathbf{m}(t)][\mathcal{X}_i(t) - \mathbf{m}(t)]^T \quad (3.50)$$

where the weights  $w_i = \frac{1}{2n} \forall i$  in terms of a symmetric sigma-point set.

### 3.6.2 Analysis and Discussion

The presented form of the unscented transformation has no design parameter, though many modifications have been developed over the past years to improve its performance e.g. [16, 17]. The concept is similar to one of the Monte Carlo simulation, but the computational costs is way below that of Monte Carlo. The speed and accuracy are considerably high.

UT delivers a second order approximation of the first two moments, but only those first two moments can be propagated which may be inadequate for some applications [3].

Note that the unscented transformation can be also used to approximate integrals of the form of Eq. (2.2) and (2.3) or some expectation values of functions of random variables such as Eq. (3.34) and (3.37) in CADET technique. In such case, the transformation part in Eq. (3.48) is skipped.

The UT can also be used in a discrete-time, modifying the transformation Eq. (3.48).

Note that the unscented transformation also appears in the continuous-time unscented Kalman filter. A graphical illustration of the UT is shown in Fig. 3.7, where the dimensionality  $n = 4$  (marginal distribution is drawn) which leads to 8 UT samples.

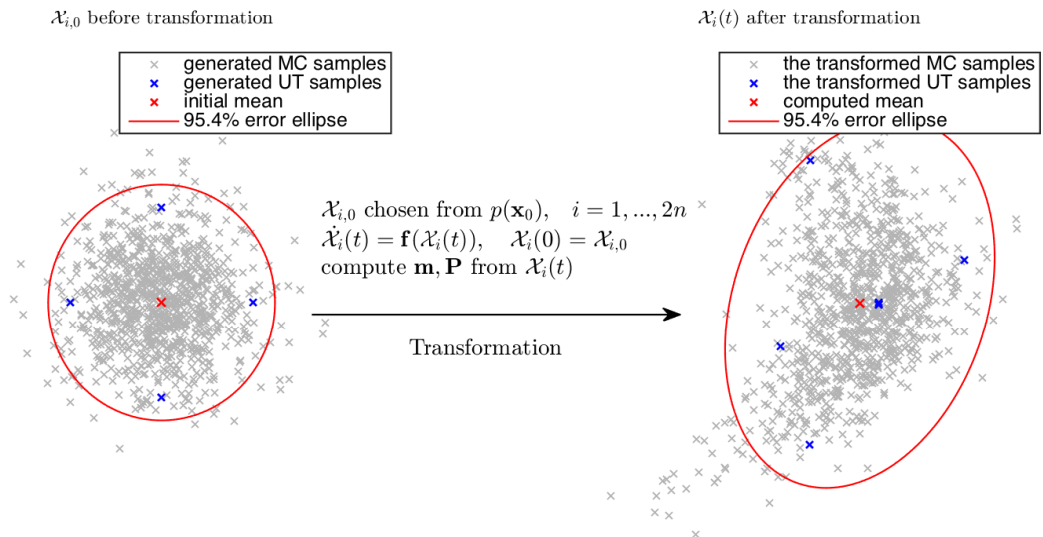


Figure 3.7: UT propagation graphical illustration, with comparison to 1000 Monte Carlo samples (gray).

## 3.7 AEGIS

The ‘‘Adaptive, Entropy-Based Gaussian-Mixture Information Synthesis’’ is a method that uses a nonlinear transformation for propagation and a Gaussian-Mixture model for uncertainty representation. In order to preserve high accuracy, AEGIS method change the number of GMM components adaptively with respect to the entropy difference between linear and nonlinear uncertainty propagation. It was first established by DeMars et al. [6].

### 3.7.1 The Original Method Description

First, assuming the initial PDF is Gaussian, the following terms are established

$$p_{\text{GMM}}(\mathbf{x}(t_0)) = \alpha_1 p_g(\mathbf{x}_0, \mathbf{m}_0, \mathbf{P}_0) \quad (3.51)$$

$$H_0 \triangleq H_1^N(t_0) = \frac{1}{2} \log |2\pi e \mathbf{P}_0| \quad (3.52)$$

where  $p_{\text{GMM}}(\mathbf{x}(t_0))$  is the initial GMM,  $\alpha_1 = 1$  is the weight of the initial component, and  $H_0$  establishes a threshold as  $\epsilon \cdot H_0$ , where  $\epsilon \in \mathbb{R}$  is a design parameter. The propagation process will be now described using following iteration.

Consider following GMM at time  $t_{s-1}$  during the time evolution,

$$p_{\text{GMM}}(\mathbf{x}(t_{s-1})) = \sum_{i=1}^l \alpha_i p_g(\mathbf{x}(t_{s-1}), \mathbf{m}_i(t_{s-1}), \mathbf{P}_i(t_{s-1})) \quad (3.53)$$

Then, each component is propagated using the UT (see Section 3.6) propagation tool starting from the time  $t_{s-1}$  until (not known yet) time  $t_s$ . The weight associated with  $i$ -th component is held constant during the propagation. Denote the resulting mean and covariance at time  $t > t_{s-1}$  as  $\mathbf{m}_i^N(t)$ ,  $\mathbf{P}_i^N(t)$  respectively. In parallel, each component entropy is monitored in terms of nonlinear entropy  $H_i^N(t)$  (the UT) and linear entropy  $H_i^L(t)$  (see Section 2.3.1) propagation via following equations,

$$H_i^N(t) = \frac{1}{2} \log |2\pi e \mathbf{P}_i(t)| \quad (3.54)$$

$$\dot{H}_i^L(t) = \text{trace } \mathbf{A}(\mathbf{m}_i(t), t), \quad H_i^L(t_{s-1}) = H_i^N(t_{s-1}), \quad t > t_{s-1} \quad (3.55)$$

If the difference between  $H_i^N(t)$  and  $H_i^L(t)$  for any  $i$  at any time  $t$  exceeds the given threshold  $\epsilon H_0$ , the propagation is halted and the  $t_s := t$ . Then, the splitting process (see Section 2.2) is applied to the  $i$ -th component,  $\alpha_i p_g(\mathbf{x}(t_s), \mathbf{m}_i^N(t_s), \mathbf{P}_i^N(t_s))$ , to yield new components which replace the  $i$ -th component,

$$p_{\text{GMM}}(\mathbf{x}(t_s)) = \sum_{i=1}^{l+\Delta l-1} \alpha_i p_g(\mathbf{x}(t_s), \mathbf{m}_i(t_s), \mathbf{P}_i(t_s)) \quad (3.56)$$

where  $\Delta l$  is the number of components the chosen library splits in. After that, the iteration is reestablished with setting  $t_{s-1} := t_s$ .

### 3.7.2 Improvements of the Method

As mentioned in Section 2.3, the differential entropy of a Gaussian distribution becomes negative if  $|\mathbf{P}| < \frac{1}{(2\pi e)^n}$  which happens if the covariance matrix  $\mathbf{P}$  is small, leading to  $p_g(\mathbf{m}) > e^{n/2}$ . If  $\mathbf{P}_0$  satisfy such property,  $H_0 < 0$  and therefore the term  $|H_i^L - H_i^N| \geq H_0$  does not make sense while  $|H_i^L - H_i^N| > 0$  for every  $t$ . This means that the GMM is split at every time  $t$ , meaning the GMM number of components increases without bounds.

A possible solution is to *scale the covariance matrix* with some parameter  $\xi > 0$  to yield a positive entropy. That can be done only within a specific application, because the design parameter  $\xi > 0$  cannot be established before having any information about the system. Taking the account of variability of the design parameter, the lack of generality may or may not be balanced by a possible improvement.

Another approach to cope with the problem is, instead of the entropy  $H_i^N(t)$  defined above, taking some chosen *function* denoted e.g.  $\mathcal{H}$  of the entropy. Consider taking the exponential function of  $H_i^N(t)$ , as

$$\tilde{H}_i^N \triangleq \mathcal{H}(H_i^N) = \exp\left\{\frac{1}{2} \log |2\pi e \mathbf{P}_i|\right\} = \sqrt{|2\pi e \mathbf{P}_i|} \quad (3.57)$$

which is certainly a nonnegative function of  $\mathbf{P}_i$ .

Another useful improvement would be defining the threshold  $\epsilon H_0$  not globally, but separately for each GMM component. Consider that the splitting process of  $i$ -th component at time  $t_{s_i}$  yielded  $\Delta l$  new components. Denote  $t_{s_{r-1}} = t_{s_i}$  the time a new  $r$ -th component was created. Now define

$$H_{0,r} \triangleq H(\mathbf{P}_r(t_{s_{r-1}})) \quad (3.58)$$

which establishes a threshold for the new  $r$ -th component as  $\epsilon H_{0,r}$ . Entropy  $H$  can be computed as  $\tilde{H}_r^N(t_{s_{r-1}})$  using Eq. (3.57).

### 3.7.3 Analysis and Discussion

The AEGIS method can approximate the true, non-Gaussian PDF very accurately, depending on the chosen  $\epsilon H_{0,i}$  for each GMM component (or  $\epsilon H_0$  globally), entropy definition, chosen splitting library, or even time discretization. This makes AEGIS suitable for nonlinear systems. The GMM approach very effectively eliminates the UT propagation drawback (propagating only the first two moments) by using more Gaussians in parallel. Instead of the UT, other nonlinear propagators can be used (such as STTs, PC, FPE). Thanks to the Gaussian nature, only the first two moments require propagating.

A penalty for a good accuracy is computational cost which is considerably high. This reveals the curse of dimensionality. The use of entropy as a measure of whether the distribution retains Gaussian or not is arguable. Referring [1], other measures e.g. Cramer von Mises metric (which uses the Mahalanobis distance) could be taken, but an analytic result of the estimate of when the uncertainty becomes non-Gaussian has not yet been answered. Moreover, AEGIS can have many design parameters which can be uncomfortable from the user point of view. Also, in order to find the matrix function  $\mathbf{A}(\mathbf{x}(t), t)$ , the derivative of the system must exist.

A graphical illustrations including the method improvements are in the Fig. 3.8 (scheme) and 3.9 (possible result).

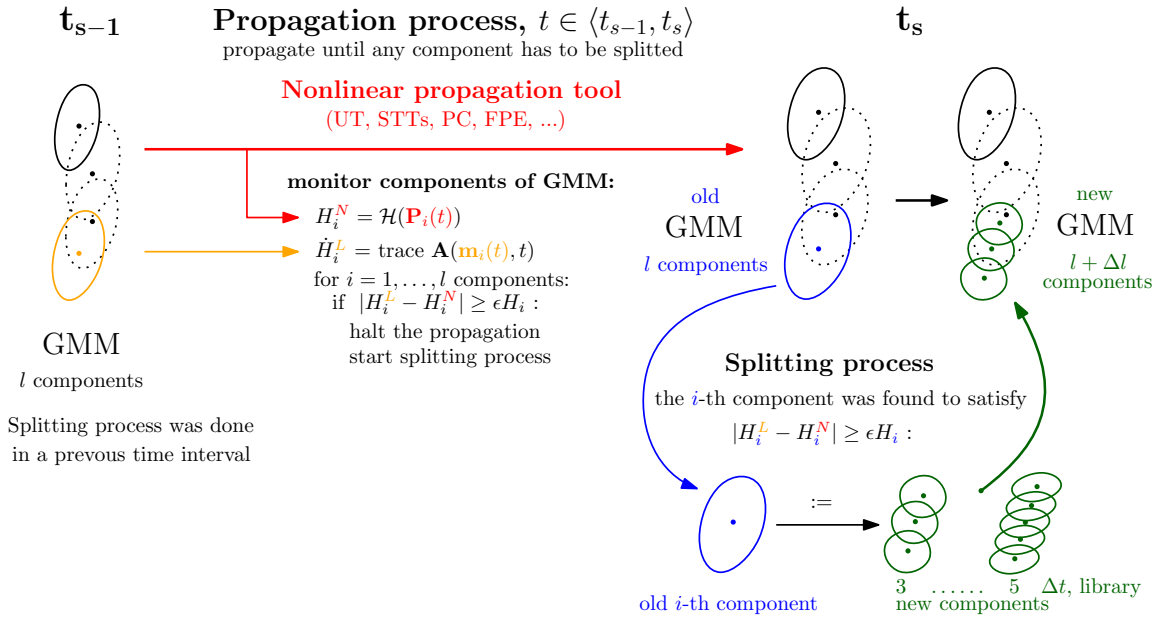


Figure 3.8: AEGIS method propagation scheme.

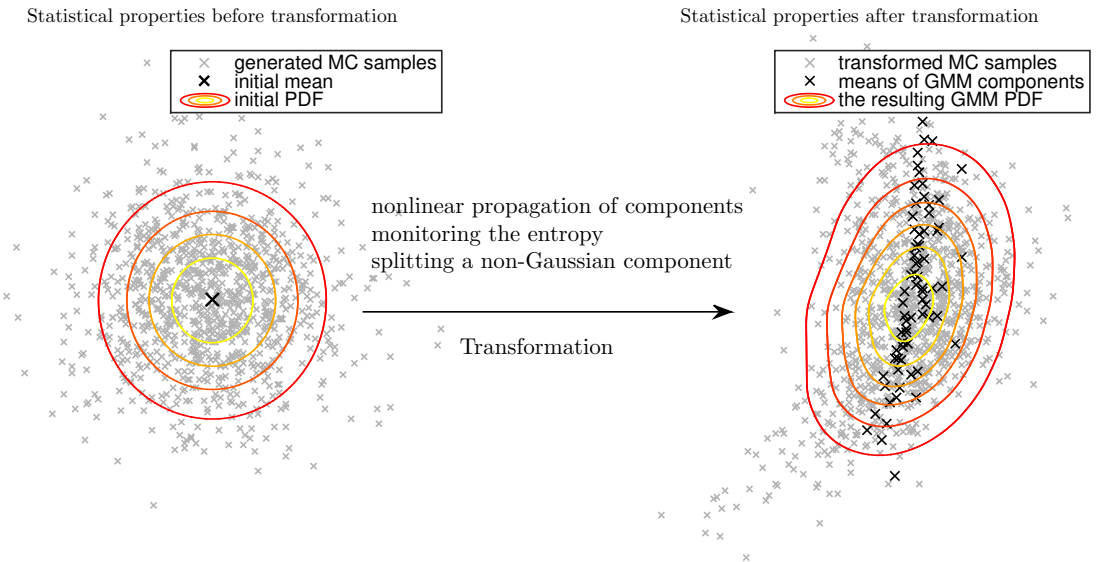


Figure 3.9: AEGIS method graphical illustration with comparison to 1000 Monte Carlo samples (gray).

---

# Testing Problems

In this chapter, two testing problems are defined, each propagation tool is then applied to the problem with following results and comparison. The testing problems should follow the main theme of this bachelor thesis, which is an uncertainty propagation for tracking of moving objects. The models should illustrate the functionality of uncertainty propagators, be intuitive and reflect the nonlinear assumption. To accomplish these objectives the two models of an orbital object movement commonly found in space object tracking [6] were chosen: an eccentric *high Earth orbit* case that is not under the influence of atmosphere, and a circular *low Earth orbit* case is under the influence of both atmospheric drag and gravity. Following the theoretically analyzed model given in Section 3.1 the testing models identify the function  $\mathbf{f}(\mathbf{x}(\mathbf{t}), \mathbf{t})$  for further simulations.

DeMars et al. in the paper [6] introduced the two models to illustrate the AEGIS method (described in Section 3.7) functionality. The other methods (stated in chapter 3) are compared in those cases in this thesis.

## 4.0.1 Likelihood Agreement Between Distributions

Performance of the methods is compared with respect to 1000 Monte Carlo Samples in chosen time steps using the likelihood agreement measure (LAM) between two distributions  $p, q$ , which is defined as

$$\mathcal{L}(p, q) = \int p(\mathbf{x})q(\mathbf{x})d\mathbf{x} \quad (4.1)$$



The LAM measures the amount of overlap between the two PDFs. The larger agreement of the two PDFs, the greater LAM. Let  $q(\mathbf{x})$  be a Dirac mixture model (DMM) of the form

$$q(\mathbf{x}) = \sum_{i=1}^N \gamma_i \delta(\mathbf{x} - \mathbf{x}_i) \quad (4.2)$$

where  $\mathbf{x}_i \forall i$  are the transformed Monte Carlo samples and  $\gamma_i = \frac{1}{N} \forall i$ . Therefore, the LAM between the DMM and Gaussian, and DMM and GMM are computed by the following equations respectively,

$$\mathcal{L}(p_g, q) = \sum_{i=1}^N \gamma_i p_g(\mathbf{x}_i, \mathbf{m}, \mathbf{P}) \quad (4.3)$$

$$\mathcal{L}(p_{\text{GMM}}, q) = \sum_{i=1}^N \sum_{j=1}^l \gamma_i \alpha_j p_g(\mathbf{x}_i, \mathbf{m}_j, \mathbf{P}_j) \quad (4.4)$$

Note, that higher value of LAM means that the given set of Monte Carlo samples are more likely to be generated by the tested PDF  $p$ .

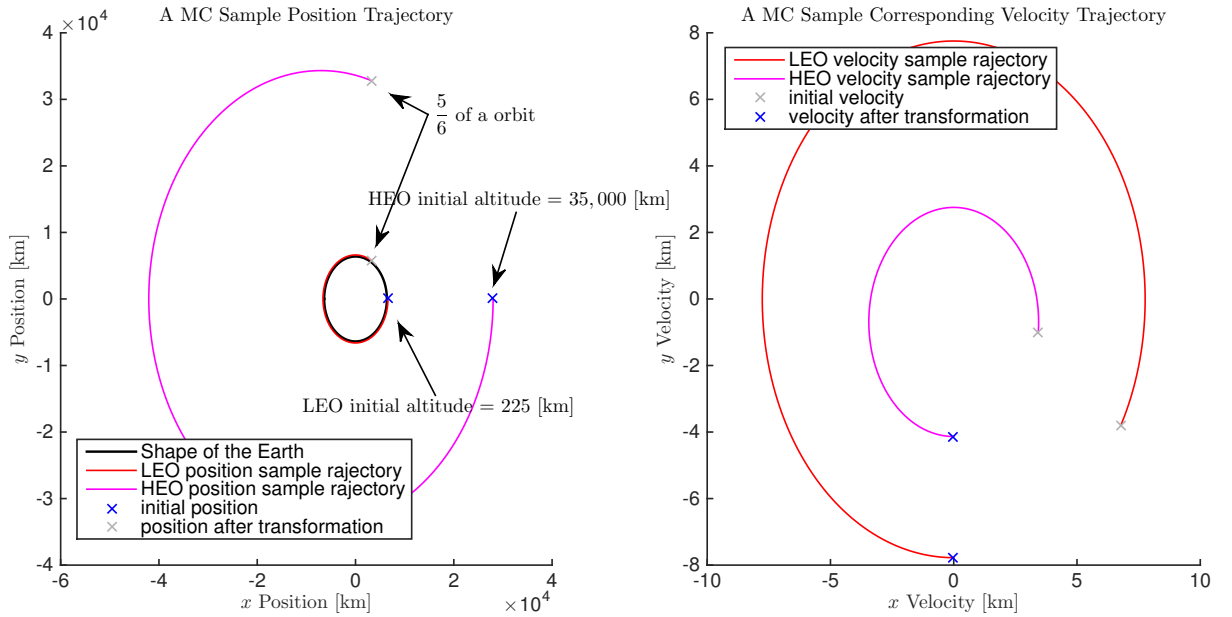


Figure 4.1: Both HEO, LEO testing problems graphical illustration from the initial state until approximately  $\frac{5}{6}$  of one orbit. Position and Velocity trajectories are taken from one of the Monte Carlo samples.

## 4.1 High-Earth-Orbit Test Case

As was mentioned before, the high Earth orbit (HEO) is eccentric and so the trajectory shape retains elliptical during time evolution. The influence of the atmosphere is not considered in this case. The governing equations of motion are taken to be

$$\dot{\mathbf{r}} = \mathbf{v}, \quad \dot{\mathbf{v}} = -\frac{\mu}{r^3}\mathbf{r} \quad (4.5)$$

where  $\mathbf{r}$ ,  $\mathbf{v}$  are the ECI position and velocity coordinates of the moving object respectively,  $r = \|\mathbf{r}\|$ ,  $\mu$  is the gravitational constant<sup>1</sup>. The motion is confined to the equatorial plane, so that the moving object position can be described with scalar values  $x = x(t)$  and  $y = y(t)$ , as well as velocity  $u = u(t)$  and  $v = v(t)$ .

The nonlinear dynamical system in form of Eq. (3.1) can than be written with

$$\mathbf{x}(t) = \begin{bmatrix} x \\ y \\ u \\ v \end{bmatrix}, \quad \mathbf{f}(\mathbf{x}(t), t) = \begin{bmatrix} u \\ v \\ -\mu x r^{-3} \\ -\mu y r^{-3} \end{bmatrix} \quad (4.6)$$

where  $r = \sqrt{x^2 + y^2}$ .

The derivative of system dynamics (Jacobian of  $\mathbf{f}(\mathbf{x}(t), t)$ ) therefore is

$$\mathbf{A}(\mathbf{x}(t)) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \mu[3x^2r^{-5} - r^{-3}] & \mu 3xyr^{-5} & 0 & 0 \\ \mu 3xyr^{-5} & \mu[3x^2r^{-5} - r^{-3}] & 0 & 0 \end{bmatrix} \quad (4.7)$$

The initial state distribution is taken to be Gaussian. The initial mean is given by Keplerian orbital elements. A semi-major axis of 35,000 km, an eccentricity of 0.2, an argument of periapse of 0 deg, and a mean anomaly of 0 deg. The initial covariance is

---

<sup>1</sup>The gravitational constant can be computed using equation  $\mu = M \cdot G$ , where  $M$  [kg] is the mass of Earth and  $G$  [ $\text{m}^3\text{kg}^{-1}\text{s}^{-2}$ ] is the Newton's gravitational constant

taken to be diagonal with position deviations of 1 km and velocity deviations of 1 m·s<sup>-1</sup>. The initial mean (transformed into Cartesian coordinates [18]) and covariance matrix both scaled into kilometers therefore are

$$\mathbf{x}_0 = \begin{bmatrix} 2.8 \times 10^4 \\ 0 \\ 0 \\ -4 \end{bmatrix}, \quad \mathbf{P}_0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 10^{-6} & 0 \\ 0 & 0 & 0 & 10^{-6} \end{bmatrix} \quad (4.8)$$

A moving object drives a single orbit after approximately 1080 minutes.

### 4.1.1 Results

Monte Carlo, LinCov, CADET and AEGIS have some design parameters. Those used within the following simulations will be therefore described now.

The number of Monte Carlo samples used is  $N = 1000$ . The LinCov time discretization parameter  $\Delta t = 1$  second was used. In CADET, the expectation values of the Eq. (3.34) and (3.37) were approximated with the UT (as suggested in Section 3.6.2). The AEGIS method is implemented twice. "s-AEGIS-3" with the 3 component splitting library; for computing the differential entropy, the scaled version  $\xi \mathbf{P}$  of covariance matrix is used with  $\xi = 10^5$  (the deviations are scaled into meters); the entropy threshold  $\epsilon H_0$  is computed using Eq. (3.52) and is same for all GMM components,  $\epsilon = 0.04$  was used. "e-AEGIS-3" with the 3 component splitting library; the entropy is computed using the exponential transformation [see Eq. (3.57)]; the entropy threshold  $\epsilon H_{0,r}$  is computed for each component separately [see Eq. (3.58)] with  $\epsilon = 0.8$ . Note that the implementation of s-AEGIS-3 should represent the original AEGIS method, while e-AEGIS-3 should represent the method improvements.

A comparison of LinCov, FOTE, CADET and UT methods with Monte Carlo simulation is shown in terms of position and velocity (marginal covariance ellipses) in Fig. 4.2 and 4.5. A comparison of s-AEGIS-3 and e-AEGIS-3 to Monte Carlo simulation in terms of position and velocity (marginal PDF contours) is shown in Fig. 4.4 and 4.3, respectively. A comparison of likelihood agreement measure of all used propagators is in Fig. 4.6. Then

the entropy  $H^L$  (linear propagator) and  $H^N$  (UT) with comparison to their transformed versions through the function  $\mathcal{H}$  are drawn in Fig. 4.7. Number of GMM components of both implementations of the AEGIS method is shown as a function of time in Fig. 4.8.

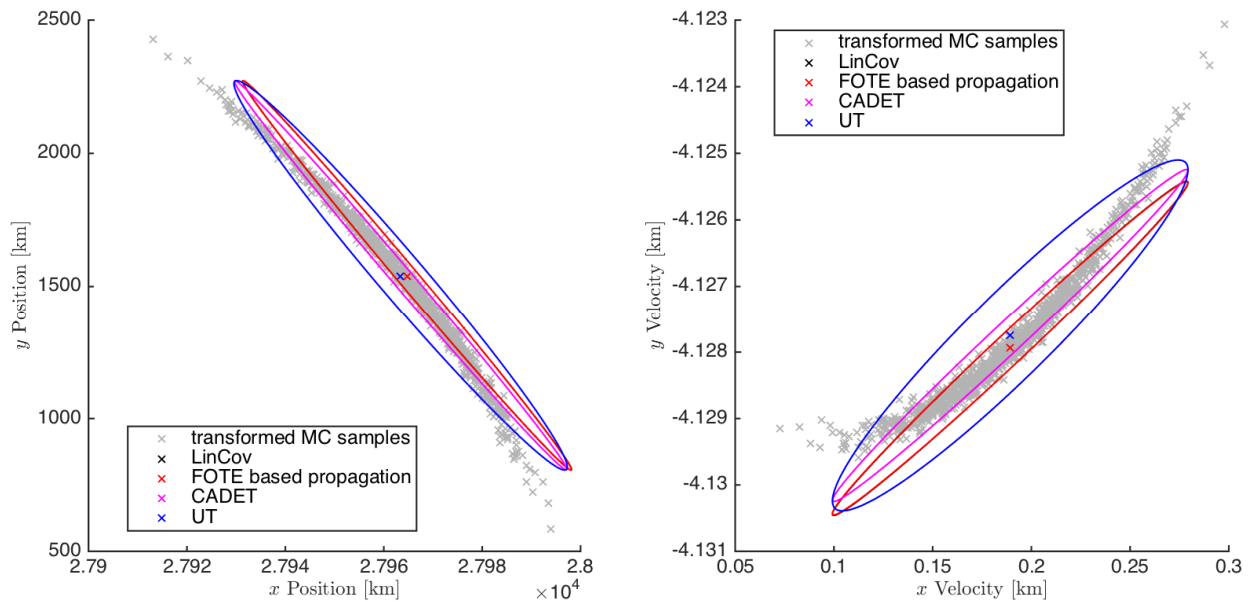


Figure 4.2: Position and Velocity results of Monte Carlo, LinCov, FOTE, CADET and UT propagators with 95.4% confidence ellipses after 18 hours (approximately one orbit).

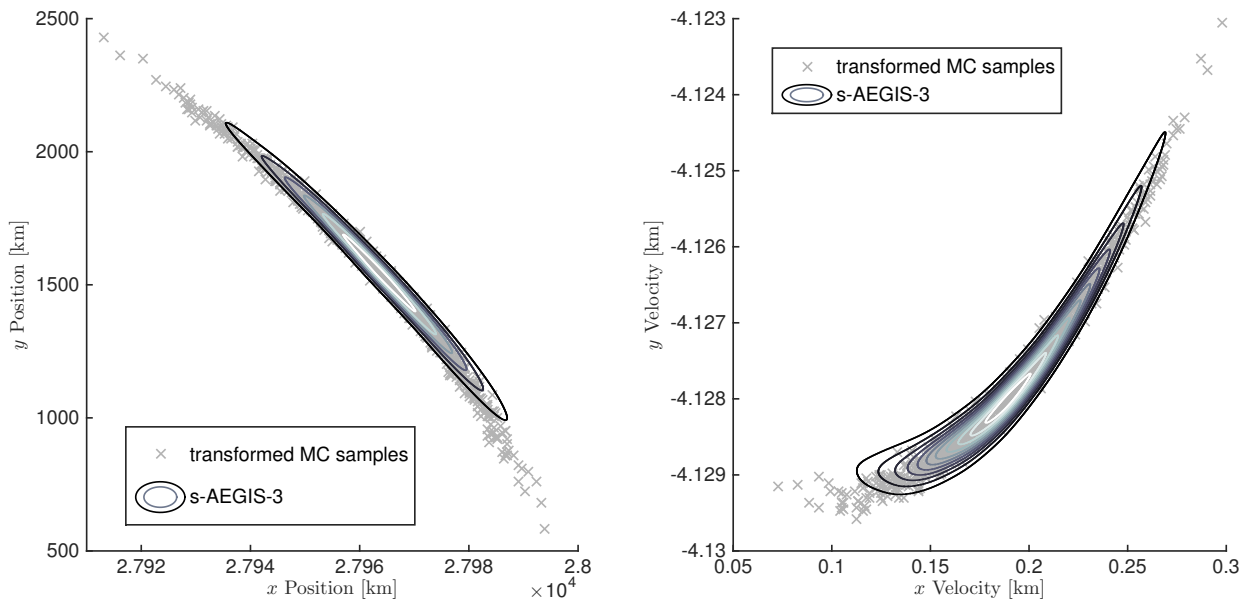


Figure 4.3: Position and Velocity marginal PDF contours of s-AEGIS-3 after 18 hours (approximately one orbit).

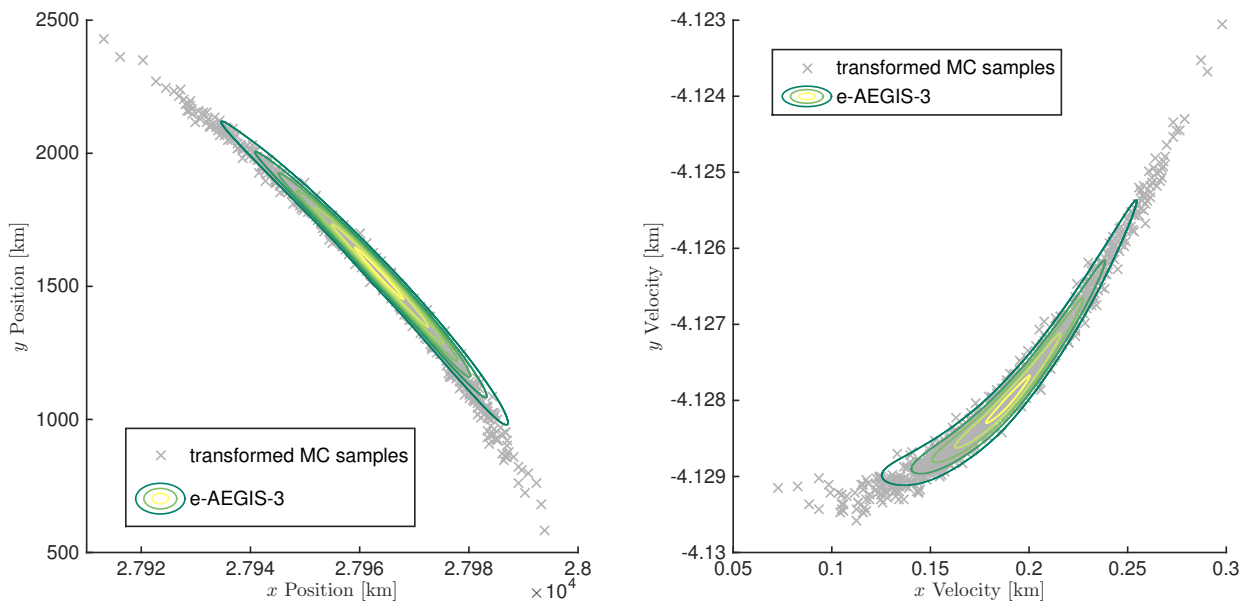


Figure 4.4: Position and Velocity marginal PDF contours of e-AEGIS-3 after 18 hours (approximately one orbit).

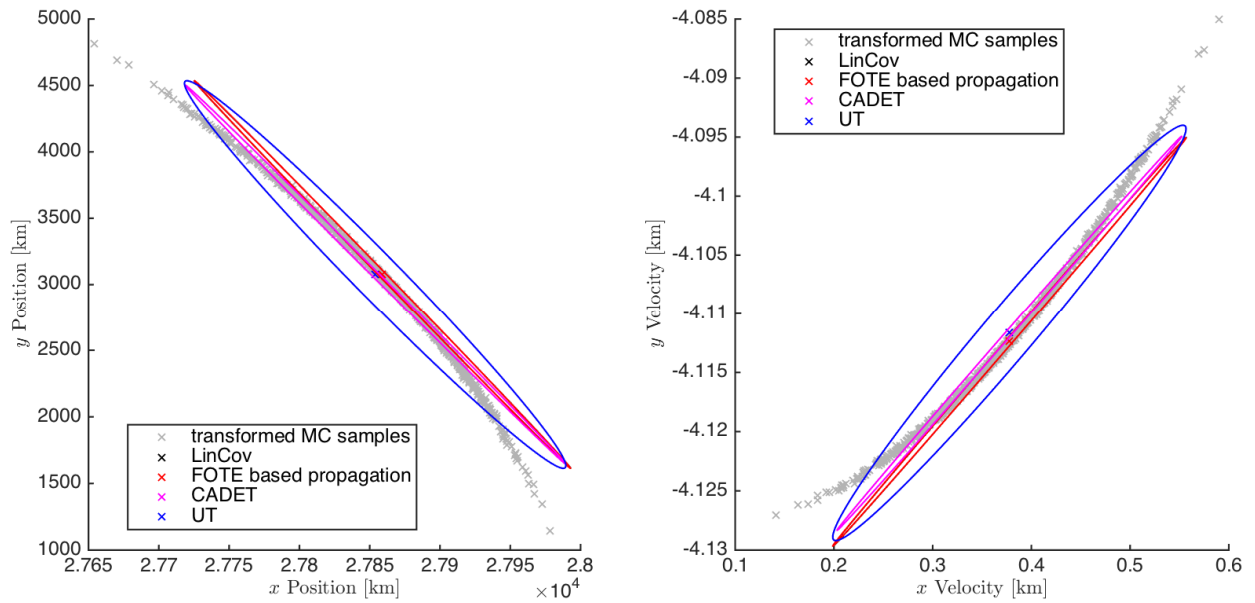


Figure 4.5: Position and Velocity results of Monte Carlo, LinCov, FOTE, CADET and UT propagators with 95.4% confidence ellipses after 36 hours (approximately two orbits).

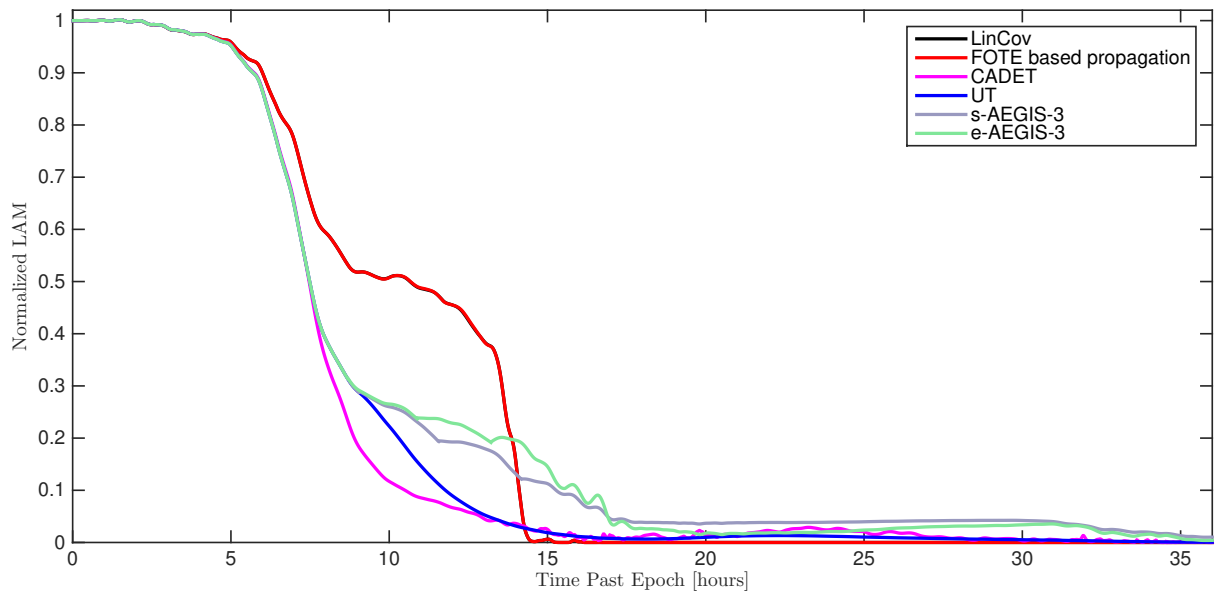


Figure 4.6: Likelihood agreement measure of LinCov, FOTE, CADET, UT, s-AEGIS-3 and e-AEGIS-3 propagators with respect to the Monte Carlo samples as a function of time, each function is normalised by LAM at the time  $t_0$ .

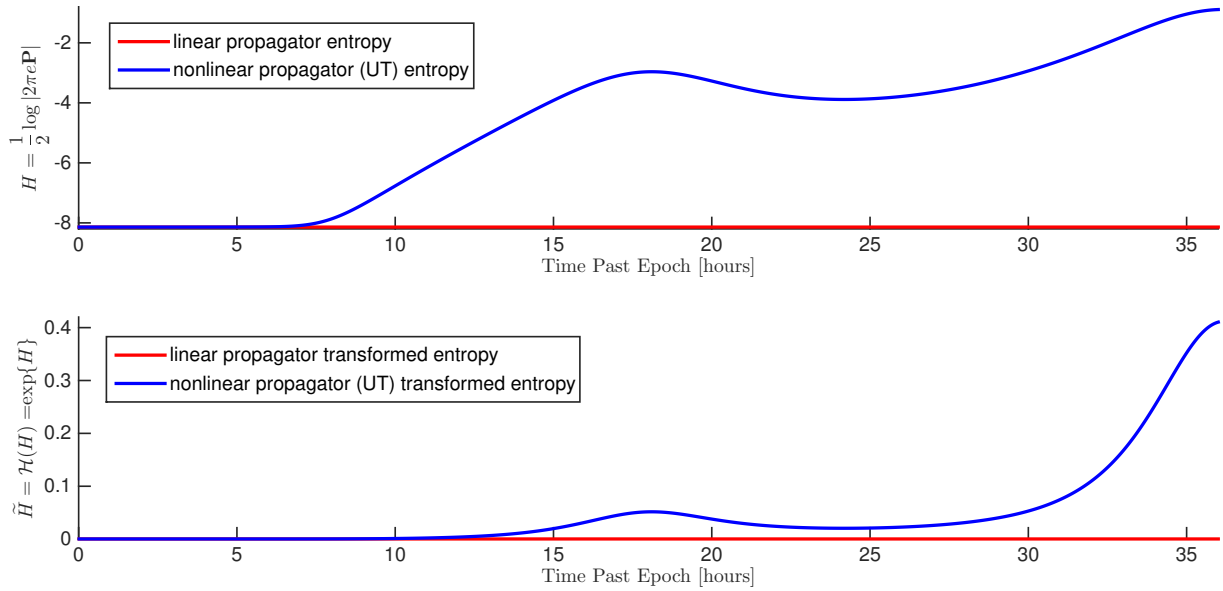


Figure 4.7: Entropy  $H(t)$  and  $\tilde{H}(t)$  of linear propagator and UT as a function of time.

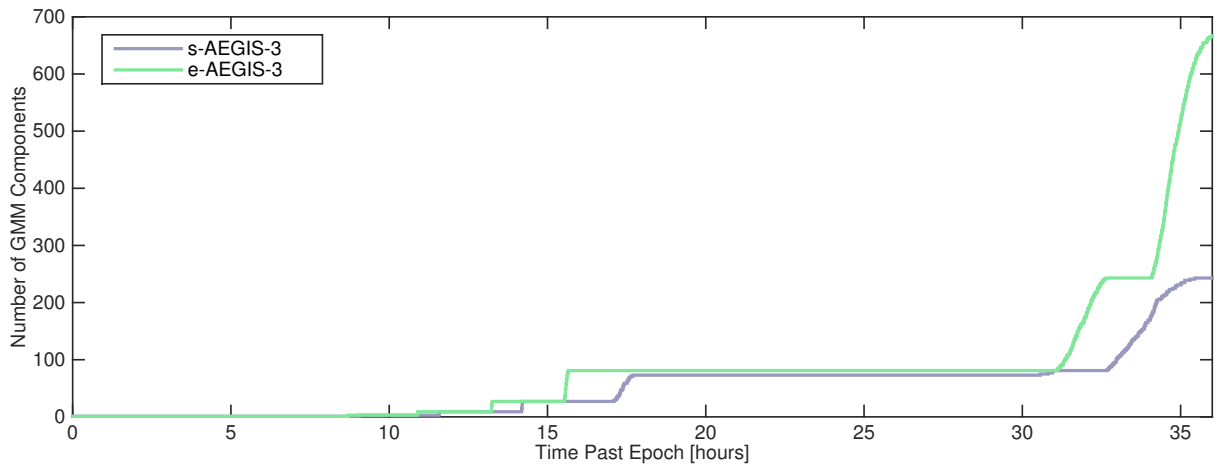


Figure 4.8: s-AEGIS-3 and e-AEGIS-3 number of GMM components as a function of time.

## 4.2 Low-Earth-Orbit Test Case

Unlike the eccentric HEO, low Earth orbit trajectory shape is circular during the time evolution. The influence of atmosphere is considered in this case. The governing equations of motion are taken to be

$$\dot{\mathbf{r}} = \mathbf{v}, \quad \dot{\mathbf{v}} = -\frac{\mu}{r^3}\mathbf{r} - \frac{1}{2}\rho(h)\beta v_{\text{rel}}\mathbf{v}_{\text{rel}} \quad (4.9)$$

where  $\rho(h)$  is the atmospheric density as a function of  $h = \sqrt{x^2 + y^2} - a$  which is the altitude of the moving object (where  $a$  is the Equatorial Earth radius taken to be 6378 km in this analysis),  $\beta$  is the ballistic coefficient (taken to be 1.4 in this analysis), and  $\mathbf{v}_{\text{rel}}$  is the inertial velocity vector (with respect to the atmosphere). Similarly to the HEO test case, the motion is confined to the equatorial plane, so that the moving object position can be described with scalar values  $x = x(t)$  and  $y = y(t)$ , as well as velocity  $u = u(t)$  and  $v = v(t)$ .

The nonlinear dynamical system in form of Eq. (3.1) can then be written with

$$\mathbf{x}(t) = \begin{bmatrix} x \\ y \\ u \\ v \end{bmatrix}, \quad \mathbf{f}(\mathbf{x}(t), t) = \begin{bmatrix} u \\ v \\ -\mu x r^{-3} - \frac{1}{2}\rho(h)\beta v_{\text{rel}}v_{\text{rel},x} \\ -\mu y r^{-3} - \frac{1}{2}\rho(h)\beta v_{\text{rel}}v_{\text{rel},y} \end{bmatrix} \quad (4.10)$$

where  $r = \sqrt{x^2 + y^2}$ ,  $v_{\text{rel},x} = u - \omega y$ ,  $v_{\text{rel},y} = v + \omega x$ ,  $\omega$  is the angular velocity of the Earth (taken to be  $7.27 \times 10^{-5}$  in this analysis), and  $v_{\text{rel}} = \sqrt{v_{\text{rel},x}^2 + v_{\text{rel},y}^2}$ . The atmospheric density is assumed to be described by an exponential atmosphere model:  $\rho(h) = \rho_0 \exp\{-\frac{h-h_0}{h_s}\}$ , where the constants in this analysis are taken to be  $\rho_0 = 3.614 \times 10^{-13} \text{ kg}\cdot\text{m}^{-3}$ ,  $h_0 = 700$  km, and  $h_s = 88.667$  km.



The derivative of system dynamics (Jacobian of  $\mathbf{f}(\mathbf{x}(\mathbf{t}), \mathbf{t})$ ) therefore is

$$\mathbf{A}(\mathbf{x}(t)) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \mu[3x^2r^{-5} - r^{-3}] + \psi_1 & \mu 3xyr^{-5} + \psi_3 & \psi_5 & \psi_7 \\ \mu 3xyr^{-5} + \psi_2 & \mu[3x^2r^{-5} - r^{-3}] + \psi_4 & \psi_6 & \psi_8 \end{bmatrix} \quad (4.11)$$

where

$$\psi_1 = \frac{1}{2}\rho(h)\beta v_{\text{rel},x} \left( \frac{x}{h_s} v_{\text{rel}} - \omega \frac{v_{\text{rel},y}}{v_{\text{rel}}} \right) \quad (4.12a)$$

$$\psi_2 = -\frac{1}{2}\rho(h)\beta \left( \frac{v_{\text{rel},x}^2}{v_{\text{rel}}} + v_{\text{rel}} \right) \quad (4.12b)$$

$$\psi_3 = \frac{1}{2}\rho(h)\beta \left( \frac{y}{h_s} v_{\text{rel}} v_{\text{rel},x} + \omega \frac{v_{\text{rel},x}^2}{v_{\text{rel}}} + \omega v_{\text{rel},x} \right) \quad (4.12c)$$

$$\psi_4 = \frac{1}{2}\rho(h)\beta v_{\text{rel},y} \left( \frac{y}{h_s} v_{\text{rel}} + \omega \frac{v_{\text{rel},x}}{v_{\text{rel}}} \right) \quad (4.12d)$$

$$\psi_5 = -\frac{1}{2}\rho(h)\beta \left( \frac{v_{\text{rel},x}^2}{v_{\text{rel}}} + v_{\text{rel}} \right) \quad (4.12e)$$

$$\psi_6 = -\frac{1}{2}\rho(h)\beta \frac{v_{\text{rel},x} v_{\text{rel},y}}{v_{\text{rel}}} \quad (4.12f)$$

$$\psi_7 = -\frac{1}{2}\rho(h)\beta \frac{v_{\text{rel},y} v_{\text{rel},x}}{v_{\text{rel}}} \quad (4.12g)$$

$$\psi_8 = -\frac{1}{2}\rho(h)\beta \left( \frac{v_{\text{rel},y}^2}{v_{\text{rel}}} + v_{\text{rel}} \right) \quad (4.12h)$$

The initial state distribution is taken to be Gaussian. The initial mean is given by an altitude of 225 km, and the initial covariance is taken to be diagonal with deviation of 1.3 km in  $x$  position, 0.5 km in  $y$  position,  $2.5 \text{ m}\cdot\text{s}^{-1}$  in  $u$  velocity, and  $5 \text{ m}\cdot\text{s}^{-1}$  in  $v$  velocity.

The initial mean and covariance matrix both scaled into kilometers therefore are

$$\mathbf{x}_0 = \begin{bmatrix} 2.603 \times 10^3 \\ 0 \\ 0 \\ -7.8 \end{bmatrix}, \quad \mathbf{P}_0 = \begin{bmatrix} 1.96 & 0 & 0 & 0 \\ 0 & 0.25 & 0 & 0 \\ 0 & 0 & 6.25 \times 10^{-6} & 0 \\ 0 & 0 & 0 & 2.5 \times 10^{-5} \end{bmatrix} \quad (4.13)$$

A moving object drives a single orbit after approximately 89 minutes.

### 4.2.1 Results

Monte Carlo, LinCov, CADET, s-AEGIS-3 and e-AEGIS-3 design parameters and/or implementation accessories retain the same as presented in Section 4.1.1. Moreover, the AEGIS method is implemented once more as "e-AEGIS-5" with the 5 component splitting library; the entropy threshold  $\epsilon H_{0,r}$  is computed for each component separately [see Eq. (3.58)] using Eq. (3.57), and  $\epsilon = 0.8$ .

A comparison of LinCov, FOTE, CADET and UT methods with Monte Carlo simulation is shown in terms of position and velocity (marginal covariance ellipses) in Fig. 4.9 and 4.10. A comparison of s-AEGIS-3, e-AEGIS-3 and e-AEGIS-5 to Monte Carlo simulation is shown in terms of position (marginal PDF contours) and velocity in Fig. 4.12, 4.11 and 4.13 respectively. A comparison of the likelihood agreement measure of all used propagators is in Fig. 4.14. Then, the entropy  $H^L$  (linear propagator) and  $H^N$  (UT) with comparison to their transformed versions through the function  $\mathcal{H}$  are drawn in Fig. 4.15. Number of GMM components of all of the implementations of the AEGIS method is compared as a function of time in Fig. 4.16.

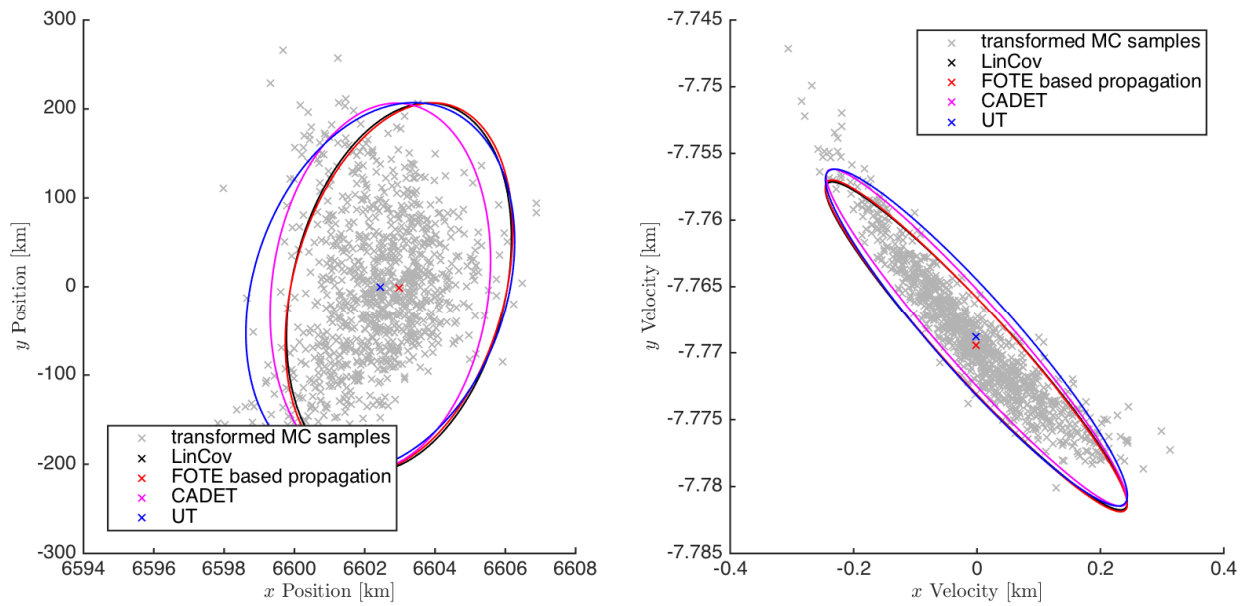


Figure 4.9: Position and Velocity results of Monte Carlo, LinCov, FOTE, CADET and UT propagators with 95.4% confidence ellipses after approximately 1.5 hours (one orbit).

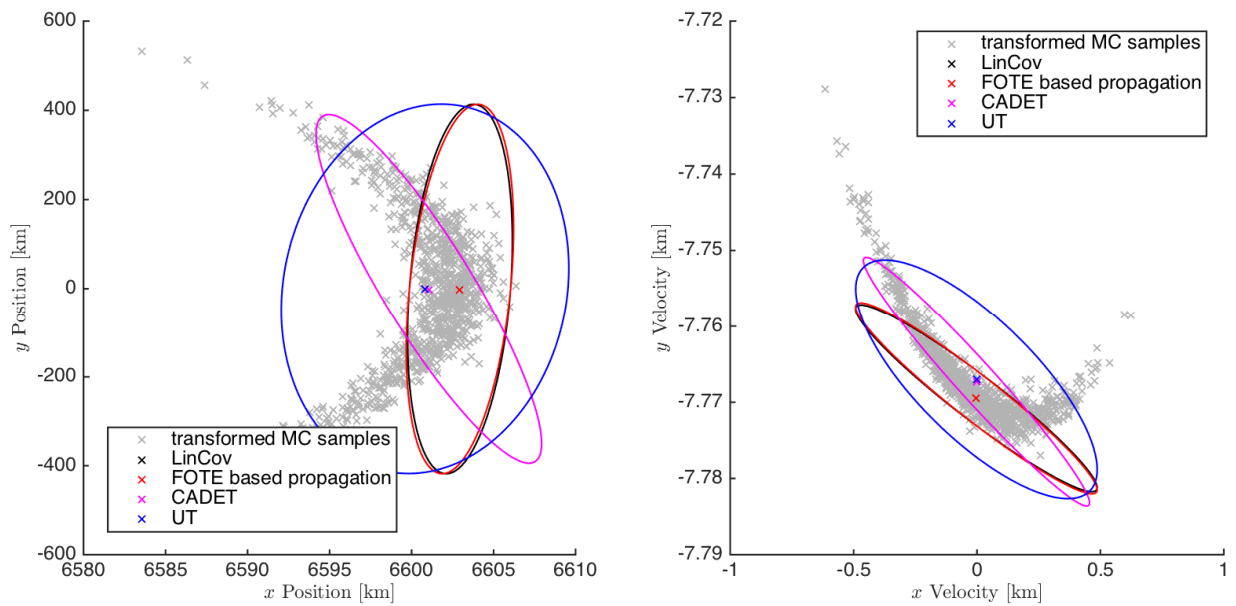


Figure 4.10: Position and Velocity results of Monte Carlo, LinCov, FOTE, CADET and UT propagators with 95.4% confidence ellipses after approximately 3 hours (two orbits).

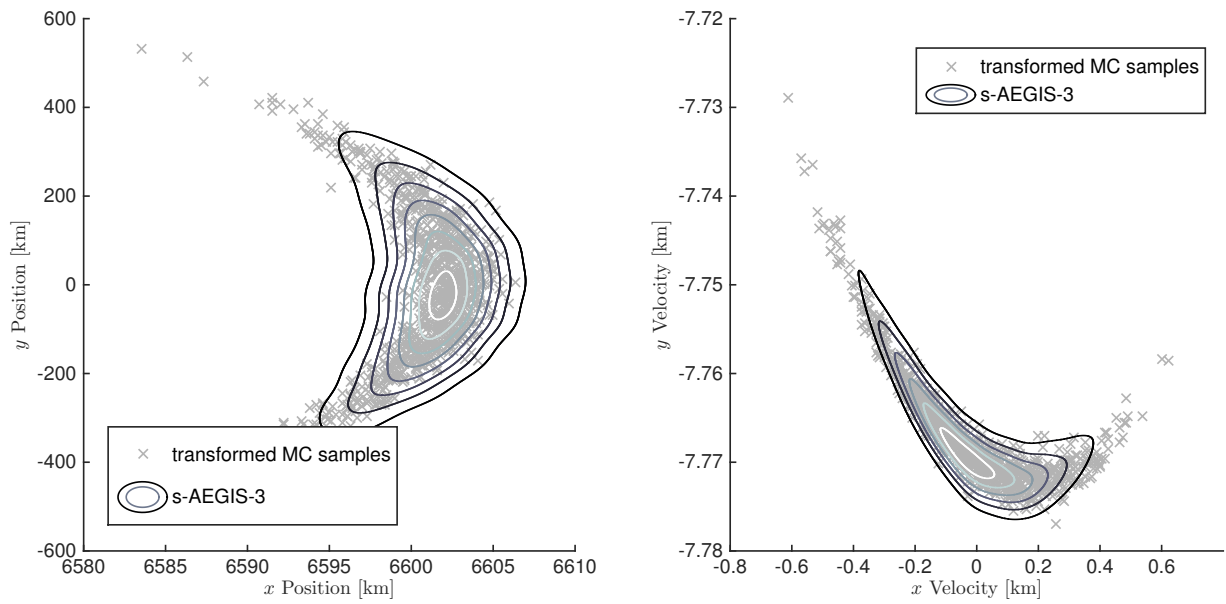


Figure 4.11: Position and Velocity marginal PDF contours of s-AEGIS-3 after approximately 3 hours (two orbits).

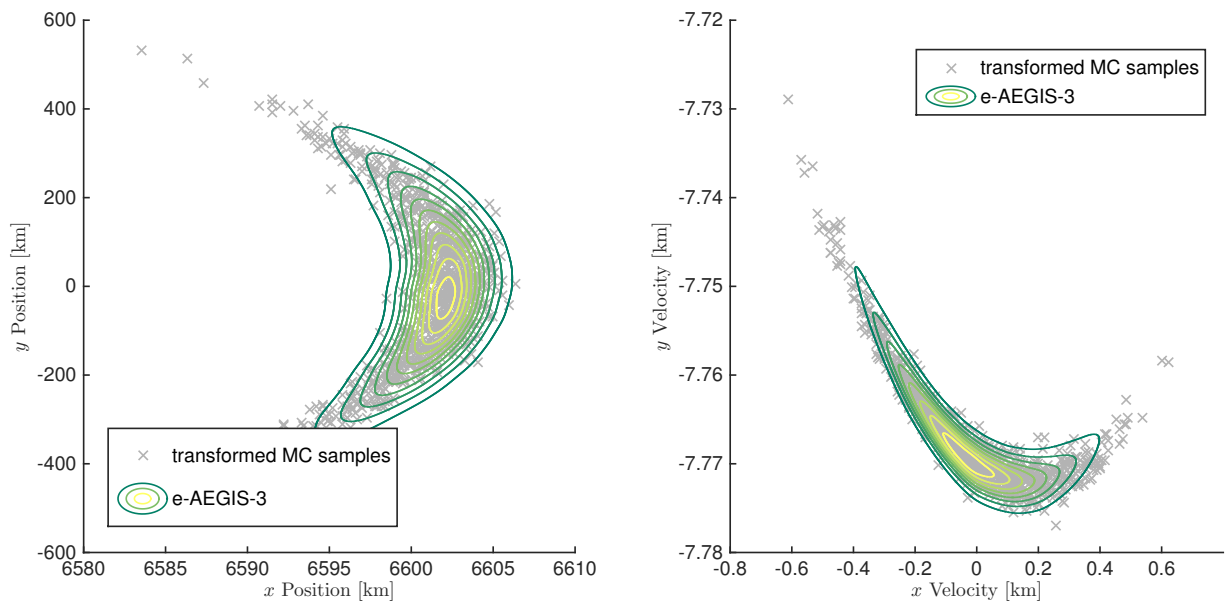


Figure 4.12: Position and Velocity marginal PDF contours of e-AEGIS-3 after approximately 3 hours (two orbits).

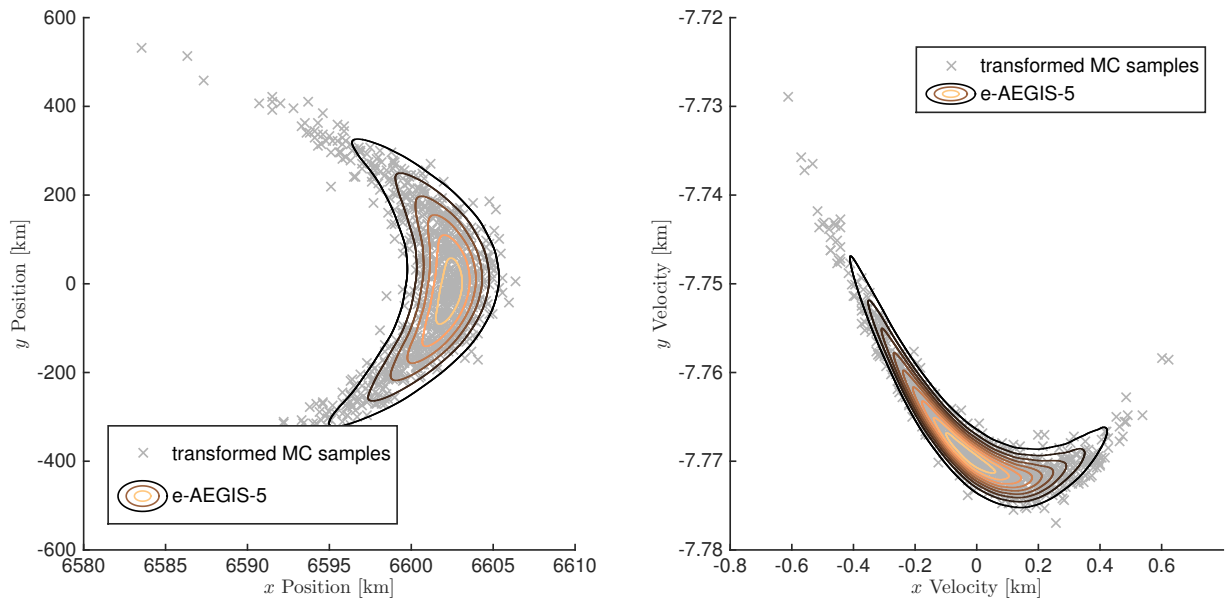


Figure 4.13: Position and Velocity marginal PDF contours of e-AEGIS-5 after approximately 3 hours (two orbits).

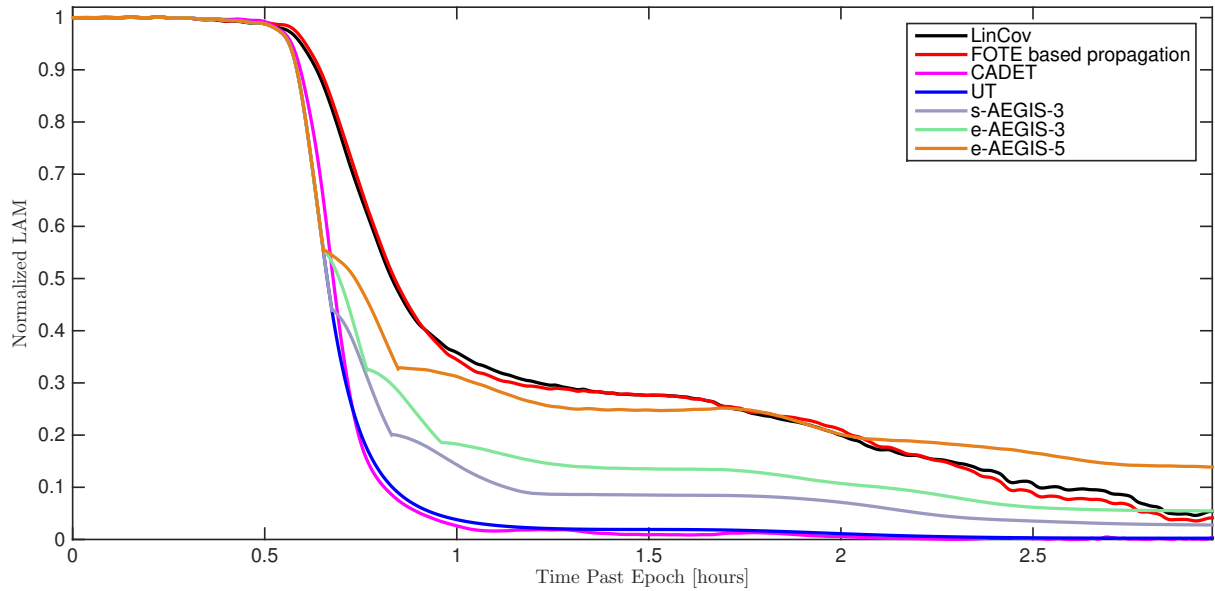


Figure 4.14: Likelihood agreement measure of LinCov, FOTE, CADET, UT, s-AEGIS-3 and e-AEGIS-3,5 propagators with respect to the Monte Carlo samples as a function of time, each function is normalised by LAM at the time  $t_0$ .

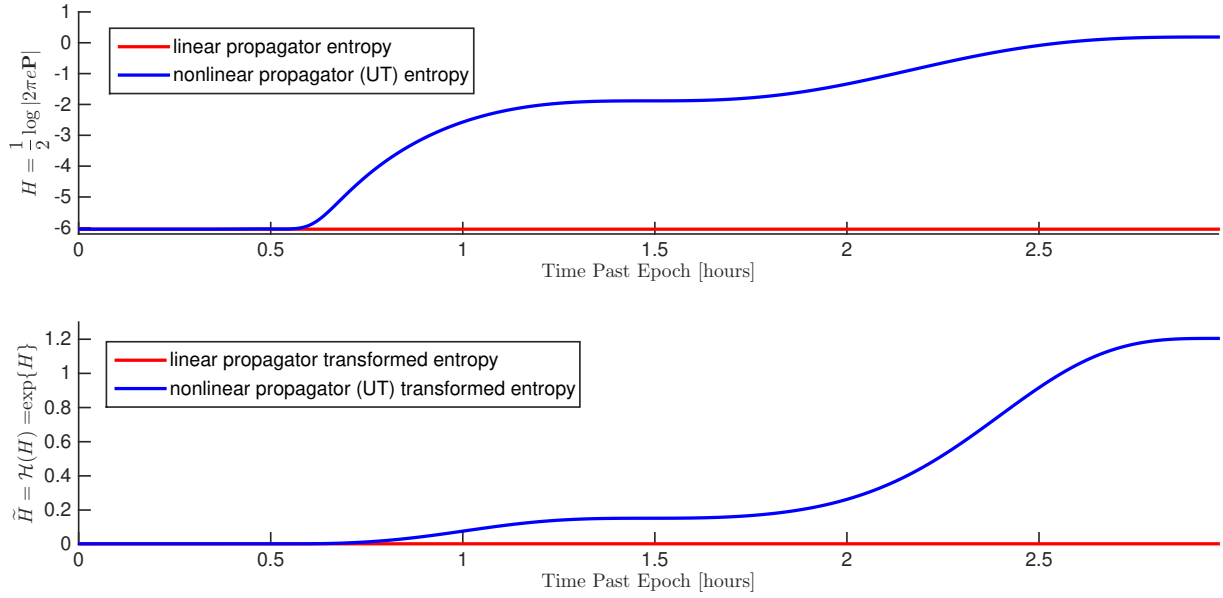


Figure 4.15: Entropy  $H(t)$  and  $\tilde{H}(t)$  of linear propagator and UT as a function of time.

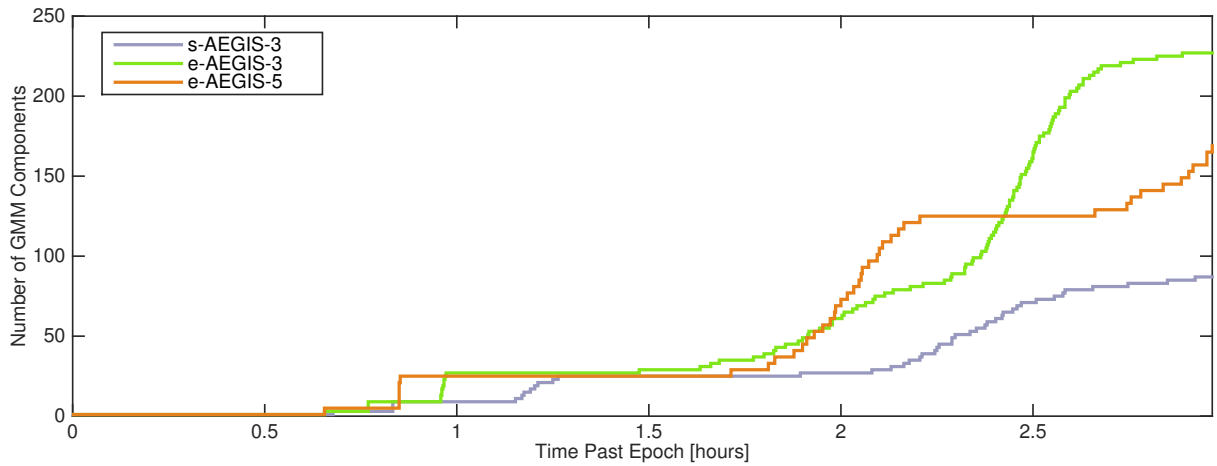


Figure 4.16: s-AEGIS-3 and e-AEGIS-3,5 number of GMM components as a function of time.

### 4.3 Analysis

The simulation results of both HEO and LEO test cases will now be discussed.

Approximate computational time costs of the implemented algorithms measured in elapsed time of the simulations<sup>2</sup> for both one and two orbit uncertainty propagation is summarized in Tab. 4.1. Note that the implementations<sup>3</sup> might execute faster, depending on the code efficiency and machine<sup>4</sup> used for simulations.

The high computational time costs of all of the implementations of AEGIS are compensated by highly accurate approximation of the true PDF shape (considered to be formed by the results of the MC simulation) which obviously cannot be reached by the LinCov, FOTE, CADET and UT propagators as they use only the first two moments for the propagation at all. This phenomenon should be supported by high accuracy measure values of the AEGIS method results, though according to the likelihood agreement measure, the AEGIS method results were only occasionally "slightly" better than the other propagation methods.

Studying the nature of the LAM, it appears that the LAM yields higher values for distributions which covariances are rather small, because the values of such PDFs near its mean are higher. This might be the case of the linear propagators. On the other hand, the UT resulting PDF obviously describe the state uncertainty more securely (with bigger covariances) than the linear methods, but its LAM values are lower than might be expected from an objective accuracy measure. Also, the UT implementation appears to have the lowest computational time costs.

Note that the entropy as a function of time (both linear and nonlinear) can be compared to the number of GMM components of the AEGIS methods, with respect to the chosen design parameters.

---

<sup>2</sup>Corresponding design parameters can be found in Section 4.1.1 and 4.2.1

<sup>3</sup>The algorithms were implemented in MATLAB<sup>®</sup> and Simulink<sup>®</sup> of version R2014b.

<sup>4</sup>The machine used for the simulations within this thesis was not real-time and had the memory of 8GB RAM and processor of 2.4 GHz, Intel<sup>®</sup> Core<sup>™</sup> i5, other processes were running during the simulations.

Table 4.1: Approximate computational time costs of the methods measured in seconds

	HEO		LEO	
	1 orbit	2 orbits	1 orbit	2 orbits
MC	6.8	9.6	6.5	10.0
LinCov	8.9	17.9	0.8	1.6
FOTE	0.2	0.2	0.2	0.2
CADET	0.6	1.4	0.3	0.8
UT	0.1	0.2	0.1	0.1
s-AEGIS-3	185.6	1 682.9	81.7	497.6
e-AEGIS-3	296.2	2 668.4	109.3	1 050.6
e-AEGIS-5	-	-	112.4	889.9



# Conclusion

Some of the currently available uncertainty propagation methods were chosen, analyzed, implemented and tested within two test cases. The methods were: Monte Carlo simulation, LinCov, first order Taylor expansion based propagation, CADET, unscented transformation and AEGIS. Also, an improvement of the AEGIS method was proposed. A possible utilization of this thesis might be as a foundation for final method selection based on specific demands in the field of uncertainty propagation (state estimation, SSA, Kalman filter framework, etc.).

The analyzed methods were applied to orbit uncertainty prediction for two cases: an eccentric high-Earth-orbit test case under the influence of gravity only and a circular low-Earth-orbit test case under the influence of both gravity and atmospheric drag. It was demonstrated how the obtained marginal PDFs (represented by contours or simply by confidence ellipses in single-Gaussian method cases) represents the curvature of the true distribution approximated by Monte Carlo simulation. Moreover, the result were compared in terms of the likelihood agreement measure to show other aspect of accuracy of each method. The most accurate prediction in sense of the true distribution curvature approximation was readily performed by the AEGIS method (any implementation) in both test cases, although its LAM was slightly worse than expected. Due to the results, the LAM might be arguable as an appropriate measure of accuracy. The approximate computational time costs were presented to get a reasonable idea of the complexity of the methods.

Future work may include analysis of methods which has not yet been compared, further improvements, combining aspects of the methods, or extending the test cases. Also, a generalized method formulation might be useful in order to meet specific demands.

---

# Bibliography

- [1] K. T. Alfriend and I. Park, “When does the uncertainty become non-gaussian,” *Advanced Maui Optical and Space Surveillance Technologies Conference*, 2016.
- [2] J. T. Horwood, N. D. Aragon, and A. B. Poore, “Gaussian sum filters for space surveillance: Theory and simulations,” *Journal of Guidance, Control, and Dynamics*, vol. 34, pp. 1839–1851, November-December 2011.
- [3] Y. zhong Luo and Z. Yang, “A review of uncertainty propagation in orbital mechanics,” *Progress in Aerospace Sciences*, vol. 89, pp. 23 – 39, 2017.
- [4] W. Oberkampf, S. M. DeLand, B. M. Rutherford, K. V. Diegert, and F. Alvin, Kenneth, “Error and uncertainty in modelling and simulation,” *Reliability Engineering and System Safety*, 2002.
- [5] H. Sorenson and D. Alspach, “Recursive Bayesian estimation using Gaussian sums,” *Automatica*, vol. 7, no. 4, pp. 465 – 479, 1971.
- [6] K. J. DeMars, R. H. Bishop, and M. K. Jah, “Entropy-based approach for uncertainty propagation of nonlinear dynamical systems,” *Journal of Guidance, Control, and Dynamics*, vol. 36, pp. 1047–1057, 2017/10/15 2013.
- [7] U. D. Hanebeck, K. Briechle, and A. Rauh, “Progressive Bayes: A new framework for nonlinear state estimation,” *Proceedings of SPIE*, vol. 5099, 2003.
- [8] C. E. Shannon, “A mathematical theory of communication,” *The Bell System Technical Journal*, vol. 27, pp. 379–423, 623–656, July, October 1948.
- [9] A. H. Jazwinski, “Stochastic processes and filtering theory,” *Dover, New York*, 126-139.

- [10] T. Caughey, “Nonlinear theory of random vibrations,” vol. 11 of *Advances in Applied Mechanics*, pp. 209 – 253, Elsevier, 1971.
- [11] O. C. Schrempf, B. Dietrich, and U. D. Hanebeck, “Density approximation based on Dirac mixtures with regard to nonlinear estimation and filtering,” *IEEE Conference on Decision and Control*, vol. 45, 2006.
- [12] F. L. Markley and J. R. Carpenter, “Generalized linear covariance analysis,” *The Journal of the Astronautical Sciences*, vol. 57, January-June 2009.
- [13] A. Gelb and R. S. Warren, “Direct statistical analysis of nonlinear systems: Cadet,” *AIAA Journal*, vol. 11, pp. 689–694, May 1973.
- [14] J. Garcke, *Sparse Grids in a Nutshell*, vol. 88 of *Lecture Notes in Computational Science and Engineering*, pp. 57–80. Springer, 2013.
- [15] S. Julier, J. Uhlmann, and H. F. Durrant-Whyte, “A new method for the nonlinear transformation of means and covariances in filters and estimators,” *IEEE Transactions on Automatic Control*, vol. 45, pp. 477–482, March 2000.
- [16] O. Straka, J. Duník, M. Šimandl, and J. Havlík, “Aspects and comparison of matrix decompositions in unscented Kalman filter,” *American Control Conference (AAC)*, 2013.
- [17] E. A. Wan and R. Van Der Merwe, “The unscented Kalman filter for nonlinear estimation,” in *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*, 2000.
- [18] R. Schwarz, “Keplerian orbital elements to cartesian state vectors,” *Memorandum no. 1, Teaching Materials*, 2017.